Image Augmentation-Based Momentum Memory Intrinsic Reward for Sparse Reward Visual Scenes

Zheng Fang, Biao Zhao, and Guizhong Liu

Abstract—Many real-life tasks can be abstracted as sparse reward visual scenes, which can make it difficult for an agent to accomplish tasks accepting only images and sparse reward. To address this problem, we split it into two parts: visual representation and sparse reward, and propose our novel framework, called image augmentation-based momentum memory intrinsic reward, which combines self-supervised representation learning with intrinsic motivation. For visual representation, we propose a representation driven by a combination of image-augmented forward dynamics and reward. To handle sparse reward, we design a new type of intrinsic reward called momentum memory intrinsic reward, which uses the difference between the outputs from the current model (online network) and the historical model (target network) to indicate the agent’s state familiarity. We evaluate our method on a visual navigation task with sparse reward in VizDoom and demonstrate that it achieves state-of-the-art performance in terms of sample efficiency. Our method is at least two times faster than existing methods and reaches a 100% success rate.

Index Terms—Agent, intrinsic motivation, reinforcement learning (RL), self-supervised learning (SSL), sparse reward visual scenes.

I. INTRODUCTION

Deep reinforcement learning (DRL) has made remarkable progress in recent years, achieving superhuman performance in various video games [1], [2], [3]. However, when we attempt to apply DRL to real-world decision-making tasks, such as navigation [4], [5] and manipulation [6], it has not attained the same astonishing performance as in video games. We believe that this is due in part to the fact that the direct models of many real tasks are sparse reward visual scenes. In other words, the agent’s observation is a high-dimensional image, and the task feedback provides positive reward only upon task completion, with zero reward at other times. This means that until the agent receives the terminal reward for the first time by chance, it has no clue about what task to accomplish. To address this issue, we propose decomposing it into two subproblems: the visual representation and the sparse reward, which hinder the training efficiency of the agent.

First, visual representation plays a critical role in DRL when the agent is required to make decisions in vision-based environments. To improve performance, one promising approach is to jointly learn a latent representation with the policy. Previous works [7], [8], [9], [10] have demonstrated that incorporating an intrinsic reward, such as self-supervised representation learning, into standard image-based DRL can lead to more robust and effective representations. However, in settings where environment rewards are sparse, these techniques may be less effective, as training a high-capacity encoder requires diverse data and dense reward feedback.

Second, sparse reward remains a significant challenge in the field of reinforcement learning (RL). The primary reason for this difficulty is that sparse reward is insufficient for enabling the agent to explore the environment thoroughly. One common method for addressing this problem is to use the intrinsic motivation exploration [11], [12] by generating intrinsic rewards. Different formulations of intrinsic rewards have been proposed, such as maximizing the visit count of less-frequently visited states [13], [14] and using prediction error as the reward signal to promote curiosity [8], [15]. However, there are two main concerns when dealing with intrinsic motivations. The first is the stochastic nature of the agent–environment system, which makes it challenging to predict accurately with a simple prediction model. The second concern is related to state representation [9]. Due to the high-dimensional property and irrelevant content of images, if intrinsic rewards are defined directly on the image, it will be difficult to express effective intrinsic motivation. Thus, a representation that can precisely capture the significant latent features of the environment state is required to distinguish novel states from previously visited ones in intrinsic reward.

We have found that both the visual representation and the sparse reward are interrelated and valuable for sparse reward visual scenes. In terms of decision-making problems, the reward-driven representation is critical, although the visual representation can be effectively enhanced through auxiliary tasks. Intrinsic motivation can address the sparse reward issue, but a well-chosen representation can make the computation of intrinsic reward more tractable and filter out irrelevant aspects of observations.

Focusing on the visual representation and the sparse reward, we have proposed the image augmentation-based momentum memory intrinsic reward (IAMMIR) framework, which efficiently fuses the self-supervised representation learning with the intrinsic motivation. In summary, this article makes the following contributions.
1) To the visual representation, we propose a self-supervised representation jointly driven by the image-augmented forward dynamics and reward. In the image-augmented forward dynamics-driven part, we operate on the image-augmented states to make predictions about future states, enabling us to extract temporal features and consistency. In the reward-driven part, we leverage a new RL objective that combines intrinsic and extrinsic rewards to guide the representation learning. Compared with prior works [7], [15], by combining both image-based and reward-based information, our approach achieves a more robust and efficient representation of visual information in the context of RL.

2) To the sparse reward, we propose a novel intrinsic reward called momentum memory intrinsic reward (MMIR), which utilizes the output error between the online network and the target network at the same state to measure the agent’s state familiarity. Unlike traditional intrinsic rewards that rely on state transition, MMIR only uses information from a single time step, avoiding interference from the stochasticity of the environment. In addition, its computation is based on the effective latent obtained from the visual representation module, ensuring the semantic expression of intrinsic rewards.

3) We demonstrate the ability of our method IAMMIR in tackling the sparse reward visual scenes, like the visual navigation tasks in VizDoom. Our method achieves at least a two times faster speed than current methods in reaching a 100% success rate, which indicates a high exploration efficiency and the state-of-the-art performance. Moreover, we also experimentally demonstrate a certain level of scene generalization with our method.

The rest of this article is organized as follows. In Section II, we introduce the related work. In Section III, we describe the problem model. In Section IV, our method is proposed in detail. In Section V, experimental results on the visual navigation tasks are shown in comparison with the prior works. Finally, Section VI concludes this article.

II. RELATED WORK

In this section, we provide a brief description on the most relevant work that our work builds on.

A. Self-Supervised Learning (SSL) in Computer Vision (CV)

In recent years, SSL has achieved great success in the CV field by learning good representations. Unlike supervised learning, it extracts training signals from a large amount of unlabeled data, which greatly improves the sample efficiency. There are mainly two types of methods in SSL, the self-predictive learning and the contrastive learning. Self-predictive learning [16], [17] refers to the paradigm in which a model learns the ability to predict a portion of input from the remaining, indirectly training the representation module. Contrastive learning [18], [19], [20], [21] directly learns a representation space where positive sample pairs are close and negative sample pairs are far apart, which provides a strong representation for the downstream tasks such as image classification. Our work is partly inspired by the method bootstrap your own latent (BYOL) [20]. We apply the same contrastive learning loss, but making corrections to the input and the network architecture to learn the temporal features.

B. Visual Representation in RL

In vision-based decision-making tasks, learning a good representation is not only beneficial for improving sample efficiency but also helps to improve decision-making performance. Therefore, there has been a substantial amount of work focused on visual representation in RL, with the most effective approach being the combination of RL and SSL. Contrastive unsupervised representations for reinforcement learning (CURL) [7] encourages the discovery of consistent features from images with SSL. Self-predictive representation (SPR) [8] learns temporal features by constructing a self-prediction auxiliary task using a transition model. More recently, image augmentations have shown significant success in learning representations. Data regularized Q (DrQ) [22] and reinforcement learning with augmented data (RAD) [10] suggest that incorporating image augmentation techniques can effectively regularize DRL from pixel inputs. Our work is also partly inspired by SPR. However, we adopt an easy forward prediction head to learn temporal features instead of a transition model. In addition to applying visual representation to policy learning, our work also applies it to the computation of the intrinsic reward.

C. Sparse Reward and Intrinsic Motivation in RL

There are lots of decision-making tasks for which the simplest way to set the reward function is the sparse reward. Yet the simplicity of sparse reward poses difficulties for RL. Intrinsically motivated exploration is an effective solution to the sparse reward problem, which can exploit various inductive biases that correlate positively with the efficient exploration. Prior works include state visitation counts [13], [14], curiosity-driven exploration [8], [15], distilling random networks [23], ensemble disagreement [24], state reachability in episodic memory [25], and so on. More recently, there has been a trend to design intrinsic reward methods with the help of self-supervised loss. Intrinsic motivation-self supervised representation (IM-SSR) [26] utilizes the different kinds of SSL loss functions as the intrinsic reward. SIM [27] also converts the self-supervised loss to an intrinsic reward to further improve generalization in RL. Not as in the previous works, we propose a novel intrinsic reward MMIR to present the agent’s state familiarity, which is based on the state representations of the present and history. Our work is similar to the concurrent work IM-SSR, which also uses SSL and intrinsic reward. However, our work just uses the output of representation module rather than the self-supervised loss, and an IM-SSR focuses on the generalization in RL, whereas our work mainly addresses the sparse reward visual scenes.

III. PROBLEM SETTING

We consider the sparse reward visual scenes as the partially observable Markov decision process (POMDP) with sparse reward setting, denoted as \((O, A, T, R, \gamma)\), where \(O\) represents a high-dimensional observation space, such as image pixels, \(A\) is
Fig. 1. We propose a representation jointly driven by the image-augmented forward dynamics and the reward. (a) In the reward driven part, the new RL objective consisting of the extrinsic and the intrinsic rewards drives the representation learning. (b) In the image-augmented forward dynamics driven part, the image-augmented states are projected into low-dimensional representations \( z_t \) and \( z_{t+1} \) through the encoders \( f_\theta \) and \( f_\xi \) and the projectors \( g_\theta \) and \( g_\xi \), and the forward dynamics head \( q_\theta \) predicts \( \hat{z}_{t+1} \) with \((z_t, a_t)\). The \( \mathcal{L}_{IFDR} \) cosine similarity loss function is used to acquire the representation.

the action space, \( T : \mathcal{O} \times \mathcal{A} \rightarrow \mathcal{P}(\mathcal{O}) \) is the observation transition probability, that is, the probability of the next observation given the last observation and action, and \( R : \mathcal{O} \times \mathcal{A} \times \mathcal{O} \rightarrow \mathbb{R} \) is the temporal reward that is defined by an observation \( o \), an action \( a \), and next observation \( o' \). In our problems, most of the rewards are 0, and only when the task is completed, a positive reward will be given. The parameter \( \gamma \in [0, 1] \) is the discount factor. To partially observable process, the general approach [28] is to stack \( k \) consecutive observations \( \{o_t-k, \ldots, o_t\} \) to represent the state \( s_t \), thereby converting the POMDP into a Markov decision problem [29] \( (S, \mathcal{A}, T, R, \gamma) \).

We seek to train a policy \( \pi(\cdot|s) : S \rightarrow \mathcal{P}(\mathcal{A}) \) whose expected cumulative discounted rewards \( \mathbb{E}_{T \sim \pi} [\sum_{t=0}^{T} \gamma^t r_t] \) are maximized in POMDP with sparse reward. To do this, we propose the IAMMIR framework, which is an easily pluggable auxiliary task that can be combined into almost any DRL algorithm to improve progressive performance and sample efficiency.

IV. METHOD

In this section, we provide technical details on IAMMIR. Specifically, we divide sparse reward visual scenes into two subproblems: visual representation and sparse reward. For these two subproblems, the IAMMIR uses self-supervised representation learning to drive the learning of state representations and intrinsic motivation exploration to solve the sparse reward problem, and effectively combines the two modules. In Section IV-A, we describe the detail of visual representation module. In Section IV-B, based on state representation, momentum memory intrinsic reward is designed.

A. Forward Dynamics- and Reward-Driven Feature

In the standard DRL, the state representation is only driven by the reward, whereas sparse reward hinders its learning in our problem. Our goal is to find more ways to drive the representation learning. Hence, we propose to jointly train an image-augmented forward dynamics representation (IFDR). Specifically, on the basis of the forward dynamics model, the input image is processed by image augmentation in advance, which can extract the temporal feature and consistency simultaneously. Our visual representation module is shown in Fig. 1, which consists of the following two parts.

1) Image-Augmented Forward Dynamics Feature: To better drive the learning of visual representations under sparse reward setting, we employ self-supervised learning to design an image-augmented forward dynamics auxiliary task. For the specific implementation details, our approach draws inspiration from BYOL [20], and the network architecture of each part is shown in the following.

**Encoder:** We use a multilayer convolution network as the encoder \( f \), as same as the architecture from intrinsic curiosity module (ICM) [15]. Specifically, each state \( s_t \) is processed by an image augmentation and the online encoder \( f_\theta \) to obtain the representation \( f_\theta(\text{aug}(s_t)) \). At the next state \( s_{t+1} \), the image augmentation is also executed. In order to prevent from the collapse of representation, the same model architecture target
encoder \( f_\xi \) is applied with its parameters updated by the exponential moving average (EMA). The EMA factor is \( \tau \in [0, 1] \)

\[
\xi \leftarrow \tau \xi + (1 - \tau)\theta.
\] (1)

A random shift and brightness transformation as the image augmentation like in the existing works is utilized [7], [8]. Kornia [30], for instance, is exploited for efficient GPU-based image augmentations.

**Projector:** The projector \( g \) is used to compress the representation output by the encoder \( f \) to a more compact feature \( z = g(f(aug(s))) \). There are online and target projectors \( g_0 \) and \( g_\xi \). The target projector parameters are given by an EMA of the online projector parameters, using the same update as the online and target encoders. Besides being refined more significant information in visual representation, the projector is also utilized as the state representation in the intrinsic reward computation to mitigate its ill effects in the new RL objective (more details in Section V-D).

**Forward dynamics head:** After the encoder \( f \) and the projector \( g \), we can get \( \hat{z}_t = g_0(f_0(aug(s_t))) \) and \( \hat{z}_{t+1} = g_\xi(f_\xi(aug(s_{t+1})) \) from the online and target network, respectively, and then design a forward dynamics task to learn the temporal features. A forward dynamics head \( q_\theta \) is trained to predict the state representations \( \hat{z}_{t+1} = q_\theta(z_t, a_t) \), which plays an important role in avoiding the representation collapse in the contrastive learning. Finally, the loss function of the IFDR is

\[
L_{\text{IFDR}}(\theta) = -\cos(\hat{z}_{t+1}, z_{t+1}).
\] (2)

2) **Reward-Driven Feature:** In the sparse reward scenes, it is hardly possible to learn a good policy from the image by using the environmental rewards. A fatal reason is that the optimization objective made up of sparse reward cannot drive the encoder to learn a sufficient state representation, which in turn affects the policy learning. To acquire better reward-driven representation, we design a new type of intrinsic reward MMIR aiming at exploring the environment efficiently. Through the combination of the intrinsic reward \( r^e \) and the extrinsic reward \( r^c \), a dense reward function \( r^{i+e} = r^e + \beta r^c \) is generated, with factor \( \beta \) reflecting the degree of exploration. The new reward function results in a new RL objective in the following:

\[
J_{\text{new}}(\theta) = \mathbb{E}_{r \sim \pi(\theta)} \left[ \sum_{t=0}^{T} \gamma^t r^{i+e}_t \right].
\] (3)

In the training process, the total loss \( L_{\text{total}}(\theta) \) includes the RL loss \( L_{\text{RL}}(\theta) \) and the IFDR loss \( L_{\text{IFDR}}(\theta) \) in (4). The IFDR loss affects the encoder \( f_0 \), the projector \( g_0 \), and the forward dynamics head \( q_\theta \). The RL loss affects the encoder \( f_\theta \) and the RL head \( p_\theta \). Our proposed method is a pluggable framework that can be applied to different RL algorithms, such as PPO [31], DDPG [32], and so on

\[
L_{\text{total}}(\theta) = L_{\text{RL}}(\theta) + \alpha L_{\text{IFDR}}(\theta).
\] (4)

**B. Momentum Memory Intrinsic Reward**

To encourage the agent to explore novel states, we design a new type of intrinsic reward to represent the agent’s state familiarity, visualized in Fig. 2. The smaller the intrinsic reward is, the more familiar the agent is with the state. We hypothesize that the difference of outputs from the current model and its historical model at the same sample can express the model’s familiarity to the samples. Similarly, we use the output error of the two networks in the visual representation module to represent the agent’s state familiarity in (5). Specifically, the current model is composed of the online encoder and the online projector \( (f_0 \circ g_0) \), and the historical model is expressed by the target network, which consists of the target encoder and the target projector \( (f_\xi \circ g_\xi) \) with the momentum update. Momentum update, also known as EMA, can be understood as the temporal ensembling of the models with exponential weights. Hence, the target network can be seen as an ensemble of the online network’s current version and those earlier versions, and the EMA coefficient \( \tau \) determines how many of the earlier versions mainly affects the target network output. For familiar states, the earlier models in the target network get the similar outputs as the online network. Whereas, for novel states, the earlier models in the target network get different results from each other, which is far away from the online network. Since the intrinsic reward is computed by two networks whose parameters have a momentum relationship, we call it MMIR

\[
r^i_t(s_t, a_t, s_{t+1}) = ||g_0(f_0(s_{t+1}) - g_\xi(f_\xi(s_{t+1}))||^2. \] (5)

In the sparse reward visual scenes, the challenge of intrinsic reward design is the uncertainty of the state transition and the meaningful state representation. As said in [8], the uncertainty of the state transition would make the prediction errors between time \( t \) and time \( t + 1 \) consistently high. In our method, however, MMIR only takes the advantage of the output error at time \( t + 1 \) of the two networks with momentum update as the intrinsic reward for time \( t \), which avoids the effects of stochastic transition. Meanwhile, the state representation is also well expressed by the representation jointly driven by the image-augmented forward dynamics and the reward.

**V. EXPERIMENT**

In this section, we verify the performance of the proposed method IAMMIR in a visual environment with sparse reward, a maze navigation scene “VizDoom” with the discrete action space, where our method plugs in the PPO [31]. Training step of all environments is less than ten million steps. We compare our method with baseline PPO, two intrinsic reward algorithms ICM [15], ECO [25], which are strong methods in VizDoom from the review [33] and the concurrent work IM-SSR [26].

**A. Environment and Setup**

VizDoom [34] provides rich maze-like 3-D environments. We test our method on two 3-D navigation task scenes. One is VizdoomMyWayHome in Fig. 3, which contains nine rooms. In this environment, the agent only accepts the image of the
Fig. 2. MMIR: We propose a new efficient and simple intrinsic reward that exploits the output error of two neural networks with momentum update. This design takes full advantage of representation learning while avoiding the uncertainty caused by state transition. The intrinsic reward \( r(s_t, a_t, s_{t+1}) \) is the L2 distance between the outputs of \( s_{t+1} \) through the online network \( f_\theta \circ g_\theta \) and the target network \( f_\xi \circ g_\xi \). For details, see Section IV-B.

Fig. 3. Top-down view, first-person view, and the terminal state from (a) the VizDoomMyWayHome and (b) VizdoomFlytrap environments. In VerySparse, the agent start at the farthest position (red point). (a) In Dense, the agent is randomly spawned in 17 locations (all points) in (a). In Flytrap, the agent start at the first room (red point). (b) The agent needs to explore the environment until it finds the armor (green point) that triggers an extrinsic reward +1.

first-person view to decide the suitable discrete action. After finding the armor, the agent receives a reward of +1, and the reward remains 0 during the rest time. An episode ends either when the agent finds the armor or when the agent has taken 2100 steps. This scene has two subscenarios: VerySparse, the agent will be spawned at the farthest position (red point). In Dense, the agent is randomly spawned in 17 locations (all points) in (a). In Flytrap, the agent start at the first room (red point). The agent needs to explore the environment until it finds the armor (green point) that triggers an extrinsic reward +1.

image of \( 4 \times 84 \times 84 \) consecutive four moments. Meanwhile, so as to speed and efficiency, we run eight environments in parallel.

B. Network Setup and Hyperparameters

The online and target encoders \( f_\theta \) and \( f_\xi \) both use the architecture from ICM. The online and target projectors \( g_\theta \) and \( g_\xi \) are two-layer MLP with a batch normalization (BN) [36] and ReLU nonlinearities, which extract latent to 256 dimensions. The predictor \( q_\theta \) is a two-layer MLP with BN and ReLU. The PPO’s RL head \( p_\theta \) contains the actor module and the critic module, both of them are a two-layer MLP with ReLU.

We train the online network parameters \( \theta \) using stochastic gradient optimization with Adam [37], where the learning rate is set to \( 2.5 \times 10^{-4} \) and mini batch size is 256. The target network parameters \( \xi \) are updated as an EMA of \( \theta \) with momentum \( \tau = 0.001 \). The factor between intrinsic and extrinsic rewards is \( \beta = 0.1 \). The coefficient \( \alpha \) in \( \mathcal{L}_{\text{total}} \) is to balance the influence between \( \mathcal{L}_{\text{PPO}} \) and \( \mathcal{L}_{\text{IFDR}} \); we set it to 0.2.

C. Navigation With Sparse Extrinsic Rewards

The average extrinsic reward curve in the period of training for the experiment is shown in Fig. 4. We can draw the following conclusions. First, our method IAMMIR can stably achieve the goal in all visual environments with sparse extrinsic rewards. Second, our method consistently beats the baseline PPO and exceeds the previous intrinsic algorithms ICM, ECO, and IM-SSR in all environments. Compared with ICM, our method is at least seven and six times faster than VerySparse and Flytrap, respectively, in reaching the 100% success rate. Compared with the ECO, our method is at least two times faster than VerySparse and Flytrap in reaching the 100% success rate. Compared with the IM-SSR, our method is always faster than VerySparse and Flytrap in reaching the 100% success rate. As far as we know, our IAMMIR is the state-of-the-art performance in VizDoom navigation task. Finally, the result in Dense shows that IAMMIR has the great scene generalization, and also gets the state-of-the-art performance in sample efficiency.
Fig. 4. The cumulative extrinsic reward curve for agent with IAMMIR, PPO, ICM, ECO, IM-SSR in Vizdoom during training. In (a) VerySparse and (b) Flytrap (extremely sparse extrinsic reward), PPO can’t learn the policy to solve the task, but after adding the intrinsic reward, the agent can solve it, and our IAMMIR is the fastest to achieve stable task completion among the intrinsic reward algorithms, with the highest data efficiency. Besides, in (c) Dense, where the agent is randomly spawned in 17 locations, our IAMMIR is still the fastest to achieve stable task completion, indicating that IAMMIR has a certain environment generalization.

Fig. 5. Average intrinsic reward curve for agent with IAMMIR in VizDoom during training. The red is MMIR with the output of projector $g$; it will decrease gradually along with train steps, indicating the process of agent’s familiarity of environment. The green is MMIR with the output of encoder $f$; it does not show a downward trend; on the contrary, it is gradually rising somewhere. (a) VerySparse. (b) Flytrap.

D. Intrinsic Reward in New RL Objective

In the new RL objective, the intrinsic reward will affect the learning of the encoder layer $f_\theta$. As discussed in previous work [38], if the representation used in the intrinsic reward is the same as the representation of the policy, the agent can artificially maximize its intrinsic reward by constructing state representations with large distances among themselves, without grounding them in environment states. As shown in Fig. 5, we find that there will be such a problem in both VerySparse and Flytrap if the output of the encoder is used directly in the MMIR. But in our work, we turn to the output of the projector $g_\theta$ to compute the MMIR. It is found that the intrinsic reward with the representation of the projector satisfies the basic hypothesis that the state familiarity value will decrease gradually along with the training time. At the same time, in theory, the intrinsic reward in the new RL objective will not affect the parameters of projector $g_\theta$, which can accurately characterize the environment states.

E. Ablation Study

To investigate the contributions of the components within the proposed method, especially, the IFDR and MMIR, we conducted an ablation analysis by the agent without IFDR or without MMIR in VerySparse. As shown in Fig. 6, the MMIR module is the most crucial component of IAMMIR, removing it caused failure in the VerySparse task. Unsurprisingly, the removal of IFDR also causes a large drop in the learning speed. The same is true, for the sparse reward visual scenes, a good representation needs to be driven by diverse data and dense reward feedback, which comes from sufficient exploration. Meanwhile, sufficient exploration requires a good representation to distinguish between the visited states and novel states. Therefore, it is a
very worthwhile solution for the sparse reward visual scenes to combine the visual representation learning with the intrinsic reward algorithms.

To verify the performance of our proposed IFDR, ablation experiments are performed on the image augmentation IA and forward dynamics prediction FDR in VerySparse. As shown in Fig. 7, first, the sample efficiency decreases greatly after removing either part of IFDR, which indicates that the consistency features and the temporal features are helpful to improve the sample efficiency. Second, in terms of the degree of the decline in the sample efficiency, the temporal features obtained by the forward dynamics prediction are more important than the consistency features obtained by image augmentation. It is also true that for sequential decision problems, figuring out the temporal mechanism of the scene is a prerequisite for decision making.

VI. CONCLUSION

To address the problem of the sparse reward visual scenes, we decompose it into two subproblems: the visual representation and the sparse reward. For the two subproblems, a novel method IAMMIR is proposed. Specifically, a sufficient state representation is acquired by the self-supervised representation learning, on the basis of which a novel momentum memory intrinsic reward MMIR is designed. We conduct experiments in three visual navigation scenes in VizDoom and show that our IAMMIR helps greatly in improving the exploration efficiency and achieves the state-of-the-art performance. We believe that IAMMIR is a superior framework for sparse reward visual scenes. We also apply our proposed framework to the environment with continuous action space like DMControl in the Appendix, and our next step is to explore more applications.

APPENDIX

PERFORMANCE IN DMControl

A. Environment and Setup

DMControl is a challenging benchmark environment for visual continuous control tasks, where agents make decisions solely based on third-person-view images. This benchmark has the following several characteristics.

1) It includes a set of challenging and diverse tasks.
2) Visual-based DRL algorithms are inefficient in sample efficiency on these benchmarks.
3) DMControl’s operation mechanism is relevant to machine manipulation in the real world.

Our algorithm is tested on DMControl’s eight tasks to evaluate its performance.

Hyperparameters of the IAMMIR are as follows. The factor between intrinsic and extrinsic rewards is $\beta = 1$. The coefficient $\alpha$ in $L^{\text{total}}$ is to balance the influence between $L^{\text{DDPG}}$ and $L^{\text{IFDR}}$; we set it to 0.5. The main compared baseline algorithms are DDPG, curiosity-driven exploration algorithm ICM, and concurrent work IM-SSR.

B. Experimental Results

In the experiments, we conduct a test every 10 000 interaction steps, and the test result is the average cumulative extrinsic reward over ten episodes. The average extrinsic reward curve in the period of testing for the experiment is shown in Fig. 8. It can be seen that the IAMMIR outperforms both the intrinsic...
reward algorithm ICM and the baseline algorithm DDPG, and is similar to the IM-SSR in terms of both sample efficiency and asymptotic performance in all eight experiments. Particularly, in the “walker_walk” and “walker_run” tasks, DDPG and ICM algorithms cannot solve these two tasks at all, while the IAMMIR can solve them well. This reflects that our IAMMIR still has high applicability in continuous action space tasks and can be applied to various baseline DRL algorithms as an easy-to-plug solution.

In these eight experiments, all are vision-based decision-making scenes, among which “pendulum_swingup”, “cart-pole_swingup_sparses”, “ball_in_cup_catch”, and “finger_spin” are sparse reward visual scenes, that is, in these tasks, only when the goal is achieved, a positive reward of 1 is given, and a reward of 0 is given at other times; the other test tasks feedback dense external rewards based on certain indicators of the agent. As shown in Fig. 8, the baseline DDPG also has good performance on these four visual sparse reward tasks, so these four tasks are considered simple sparse reward visual tasks. In order to demonstrate the superiority of our IAMMIR, the reward function of the “walker_walk” and “hopper_hop” tasks is modified to be sparse reward visual scenes, as shown in (6). The final experimental results are shown in Fig. 9. It can be seen from the figure that our IAMMIR can still outperform the baseline algorithm in terms of sample efficiency and asymptotic performance in more difficult sparse reward visual scenes.

\[
r' = \begin{cases} 
1 & r \geq 0.5 \\
0 & r < 0.
\end{cases}
\] (6)

REFERENCES

[1] O. Vinyals et al., “Grandmaster level in StarCraft II using multi-agent reinforcement learning,” Nature, vol. 575, no. 7782, pp. 350–354, 2019.
[2] C. Berner et al., “Data-efficient reinforcement learning with self-predictive representations for autonomous mental development,” IEEE Trans. Evol. Comput., vol. 11, no. 2, pp. 265–286, Apr. 2007.
[3] A. P. Badia et al., “Agent57: Outperforming the Atari human benchmark,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 507–517.
[4] L. Tai, G. Paolo, and M. Liu, “Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2017, pp. 31–36.
[5] O. Zhelo, J. Zhang, L. Tai, M. Liu, and W. Burgard, “Curiosity-driven exploration for mapless navigation with deep reinforcement learning,” vol. abs/1804.00456, 2018. [Online]. Available: https://arxiv.org/abs/1804.00456
[6] P. Sermanet et al., “Time-contrastive networks: Self-supervised learning from video,” in Proc. IEEE Int. Conf. Robot. Autom., 2018, pp. 1134–1141.
[7] M. Laskin, A. Srinivars, and P. Abbeel, “CURL: Contrastive unsupervised representations for reinforcement learning,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 5639–5650.
[8] M. Schwarzer, A. Anand, R. Goel, R. D. Hjelm, A. C. Courville, and P. Bachman, “Data-efficient reinforcement learning with self-predictive representations,” in Proc. 9th Int. Conf. Learn. Representations, 2021. [Online]. Available: https://openreview.net/forum?id=ucQfPZwRaU
[9] D. Yarats, R. Fergus, A. Lazaric, and L. Pinto, “Reinforcement learning with prototypical representations,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 11920–11931.
[10] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivars, “Reinforcement learning with augmented data,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 19884–19895, 2020.
[11] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, “Intrinsic motivation systems for autonomous mental development,” IEEE Trans. Evol. Comput., vol. 11, no. 2, pp. 230–247, Sep. 2010.
[12] J. Schmidhuber, “Formal theory of creativity, fun, and intrinsic motivation (1990–2010),” IEEE Trans. Auton. Ment. Develop., vol. 2, no. 3, pp. 230–247, Sep. 2010.
[13] M. G. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, “Unifying count-based exploration and intrinsic motivation,” in Proc. Adv. Neural Inf. Process. Syst. 29; Annu. Conf. Neural Inf. Process. Syst., D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., Barcelona, Spain, vol. 29, Dec. 5-10, 2016, pp. 1471–1479.
[14] G. Ostrovski, M. G. Bellemare, A. Oord, and R. Munos, “Count-based exploration with neural density models,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 2721–2730.
[15] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, “Curiosity-driven exploration by self-supervised prediction,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 2778–2787.
[16] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, “Tracking emerges by colorizing videos,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 391–408.
[17] X. Wang, A. Jabri, and A. A. Efros, “Learning correspondence from the cycle-consistency of time,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2019, pp. 2566–2576.
[18] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 1597–1607.

[19] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 9729–9738.

[20] J.-B. Grill et al., “Bootstrap your own latent—A new approach to self-supervised learning,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 21271–21284, 2020.

[21] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, “Unsupervised learning of visual features by contrasting cluster assignments,” Adv. Neural Inf. Process. Syst., vol. 33, pp. 9912–9924, 2020.

[22] D. Yarats, R. Fergus, A. Lazaric, and L. Pinto, “Mastering visual continuous control: Improved data-augmented reinforcement learning,” 2021, arXiv:2107.09645.

[23] Y. Burda, H. Edwards, A. J. Storkey, and O. Klimov, “Exploration by random network distillation,” in Proc. 7th Int. Conf. Learn. Representations, 2019. [Online]. Available: https://openreview.net/forum?id=H1JJnR5Ym

[24] D. Pathak, D. Gandhi, and A. Gupta, “Self-supervised exploration via disagreement,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 5062–5071.

[25] N. Savinov et al., “Episodic curiosity through reachability,” in Proc. 7th Int. Conf. Learn. Representations, 2019. [Online]. Available: https://openreview.net/forum?id=SkeK3s0qKQ

[26] Y. Zhao, C. Du, H. Zhao, and T. Li, “Intrinsically motivated self-supervised learning in reinforcement learning,” in Proc. Int. Conf. Robot. Automat., 2022, pp. 3605–3615.

[27] K. Wu, M. Wu, Z. Chen, Y. Xu, and X. Li, “Generalizing reinforcement learning through fusing self-supervised learning into intrinsic motivation,” in Proc. AAAI Conf. Artif. Intell., 2022, pp. 8683–8690.

[28] V. Mnih et al., “Playing Atari with deep reinforcement learning,” 2013, arXiv:1312.5602.