# Generative Model-based Exploration for Sample-Efficient Reinforcement Learning

Xinyan Cai[1,2], Shuwei Wang[1,2], Qiang Guan[1,2]
[1]University of Chinese Academy of Sciences
[2]Institute of Automation, Chinese Academy of Sciences

***Abstract*—Sample efficiency is a critical challenge in deep reinforcement learning (DRL), driving the development of numerous exploration strategies. However, efficient exploration in high-dimensional observation spaces remains difficult. To address this, we introduce Generative Model-based Exploration (GME), a novel algorithm grounded in world models. Specifically, we characterize state transitions as transformations among latent probability distributions, upon which we learn a generative world model. Our exploration term is defined using mutual information and epistemic uncertainty derived from this model. Experimental results demonstrate that GME substantially enhances the sample efficiency of baseline methods, yielding superior performance and robust generalization in unseen environments. Furthermore, extensive experiments validate that the environmental dynamics insights provided by the world model significantly facilitate exploration.**

## I. Introduction

Deep reinforcement learning (DRL) is a powerful framework for solving complex tasks [1]–[3]. However, real-world scenarios often lack dense rewards, rendering standard RL methods ineffective.

Many algorithms have been developed to assist agents in exploration by measuring the exploratory value of states [4]–[7]. These can be broadly categorized into: **Count-based Exploration:** These methods measure state novelty by tracking visitation counts, encouraging agents to visit novel states [5], [8]–[12]. **Curiosity-based Exploration:** These methods encourage agents to explore "hard-to-predict" states via curiosity modules [7], [13]–[15]. **Information-Theoretic Exploration:** These methods assess novelty by estimating the distribution entropy of trajectories. For example, MEPOL [16] estimates state entropy using k-nearest neighbors, while RE3 [17] uses a randomly initialized encoder.

Recent advancements have significantly enhanced world models [18]–[20], enabling them to effectively capture physical dynamics and predict future states [21]. Accurate world modeling allows planning algorithms to search for optimal solutions [22] or train agents directly within the model [23], [24]. This raises the question: *Can world models enhance exploration efficiency for reinforcement learning agents?*

While world models offer advantages, leveraging them for exploration poses challenges. First, inherent stochasticity—rooted in noisy observations and transition uncertainties—complicates modeling and exploration signal definition. Traditional latent representations often struggle to disentangle and explicitly capture such uncertainty. Second, learning a world model for exploration is difficult; the model must comprehend state uncertainty and transformations between stochastic states. Third, effectively quantifying novelty or information gain in latent space remains non-trivial.

In this study, we propose Generative Model-based Exploration (GME). We abstract environmental dynamics into transitions between latent distributions, with concrete observations sampled from these distributions. We learn a stochastic world model and develop an information-gain-based exploration algorithm. Unlike methods measuring novelty via visitation counts or prediction errors, GME reformulates exploration as distributional evolution in latent space, enabling explicit uncertainty quantification. We evaluate our approach across diverse environments and explore its generalizability. Our primary contributions are:

- We define the Stochastic Latent Markov Decision Process (Stochastic Latent-MDP), a hierarchical MDP that abstracts state transitions as changes between latent distributions.
- We design a discrete vector distribution to represent latent distributions and a generative world model that integrates multiple components, introducing intrinsic rewards to facilitate efficient exploration.
- GME outperforms state-of-the-art exploration algorithms in high-dimensional environments and demonstrates generalization to unseen environments and tasks.

## II. Related Work

### A. Intrinsic Reward Shaping

Achieving an appropriate balance between exploration and exploitation remains a long-standing challenge in reinforcement learning (RL) [25]. When external rewards are sparse, intrinsic rewards are often employed to encourage agents to explore [26]–[29]. The Intrinsic Curiosity Module (ICM), proposed by [13], utilizes prediction error to facilitate exploration and serves as a cornerstone for subsequent research. [14] developed an integrated set of forward models, leveraging the predictive variance of these models to represent the unpredictability of the environment. Additionally, [30] modeled actions as a latent variable $z$ and introduced a Variational Autoencoder to model both forward and backward processes, achieving expert-level performance in imitation learning. [15] proposed a Reward Influence Driven Exploration (RIDE) method that

Corresponding author: qian.guan@ia.ac.cn

uses the differences between two consecutive encoded states as intrinsic rewards, encouraging agents to select actions that lead to significant state changes. There are also approaches focused on the automatic selection of intrinsic rewards within agent learning [31]. Overall, the modeling of intrinsic rewards continues to be a promising area of exploration.

### B. World Models in RL

World models are trained to model the environment's state transition functions and are widely utilized in model-based reinforcement learning (RL) [25]. Numerous exploratory works also summarize agent experiences in the form of world models [13], [15]. A common approach involves learning an encoder that maps original observations to a latent space, with the learned transition function also based on this latent representation. Depending on the task, different world models often rely on varying latent spaces. For instance, the Dreamer series [23], [24] learns the latent space by reconstructing comprehensive environmental information, including observations, rewards, and discounts, while TD-MPC [22] focuses on task-specific information by reconstructing only rewards. In exploratory work, a prevalent method is to use an inverse dynamics model to reconstruct actions for learning dynamics-specific information [12]–[15].

### C. Uncertainty in RL

Many methods utilize uncertainty as an intrinsic reward, offering high returns when agents explore regions of the environment with high uncertainty [32]. However, modeling uncertainty in reinforcement learning (RL) presents significant challenges. [14] summarizes sources of randomness in three key areas: noisy environmental observations (e.g., background noise from a television), variability in agent-environment interactions, and randomness in the state transitions themselves.

Techniques for modeling uncertainty typically include Bayesian networks, model ensembles, and information theory. For instance, the Disagreement approach [14] employs multiple model ensembles to compute prediction variance as a measure of uncertainty. Variational Information Maximization Exploration (VIME) [26] leverages information gain as an intrinsic reward, specifically using the Kullback-Leibler (KL) divergence between the weight distributions of a Bayesian neural network before and after new data is observed. Additionally, GIRIL [30] introduces a Variational Autoencoder (VAE) to encode actions as latent variables.

## III. Background

The standard Markov Decision Process (MDP) [33] is defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{T}, \pi, r)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ is the transition function specifying the probability distribution over next states given the current state and action, and $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ is the agent's stochastic policy. At each time step $t$, the agent in state $s_t$ selects an action $a_t$ by sampling from $\pi(\cdot|s_t)$, transitions to state $s_{t+1}$ according to $\mathcal{T}(s_{t+1}|s_t, a_t)$, and receives a reward defined as $R_t = r(s_t, a_t)$. The agent's objective is to learn a policy $\pi$ that maximizes the discounted expected return within an episodic setting:

$$\mathbb{E}_\pi \left[ \sum_{k=0}^{T} \gamma^k R_{t+k} \right], \tag{1}$$

where $\gamma \in [0, 1]$ is the discount factor, $T$ is the episode horizon, and the expectation is taken over the policy $\pi$ and environment dynamics $\mathcal{T}$. Departing from this standard formulation, this paper considers dual reward sources where the total reward $R_t = r_t^e + \omega_i^r r_t^i$ comprises an extrinsic reward $r_t^e$ provided by the environment and an intrinsic reward $r_t^i$ generated internally for each transition tuple $(s_t, a_t, s_{t+1})$.

## IV. Method

We first introduce the Stochastic Latent-MDP, then present our state representation, learning objectives, and world model architecture. Finally, we detail our exploration strategy.

### A. Stochastic Latent-MDP

The **Stochastic Latent-MDP** formalizes environment dynamics as measure transformations. Formally, given observation space $\mathcal{S}$ and action space $\mathcal{A}$, we define the tuple $\langle \mathcal{P}(\mathcal{Z}), \mathcal{A}, \mathcal{T} \rangle$ where $\mathcal{P}(\mathcal{Z})$ is the latent state distribution space and $\mathcal{T} : \mathcal{P}(\mathcal{Z}) \times \mathcal{A} \to \mathcal{P}(\mathcal{Z})$ is the *stochastic transition operator* evolving probability measures.

Unlike classical MDPs that model state transitions as deterministic mappings $s_{t+1} = f(s_t, a_t)$, our framework captures fundamental environmental uncertainty through *distributional evolution*:

$$\mathcal{P}(\mathcal{Z})_{t+1} = \mathcal{T}\big(\mathcal{P}(\mathcal{Z})_t, a_t\big)$$

This enables: (1) physical grounding of actions as probability modifiers (e.g., closing a door reduces threat probability), and (2) native representation of stochastic transitions as measure transformations. The latent mapping $\mathrm{M} : \mathcal{S} \to \mathcal{X}$ abstracts observations into probabilistic latent states $\mathcal{X} \sim \mathcal{P}(\mathcal{X})$, where actions induce distribution shifts. This paradigm naturally encodes partial observability and enables direct uncertainty quantification through distribution statistics.

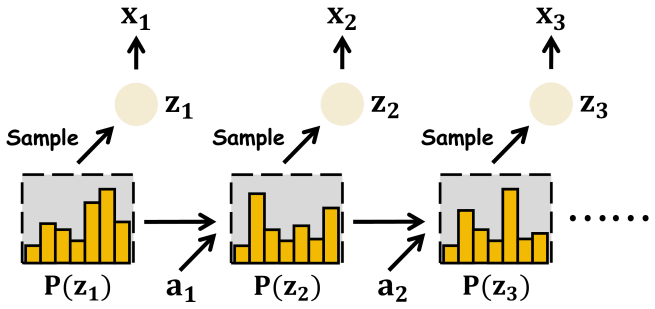



Fig. 1: Illustration of Stochastic Latent-MDP. We model the probability distribution of latent variables and interpret actions as transitions between latent states; actual latent variables and observations are sampled from this distribution.

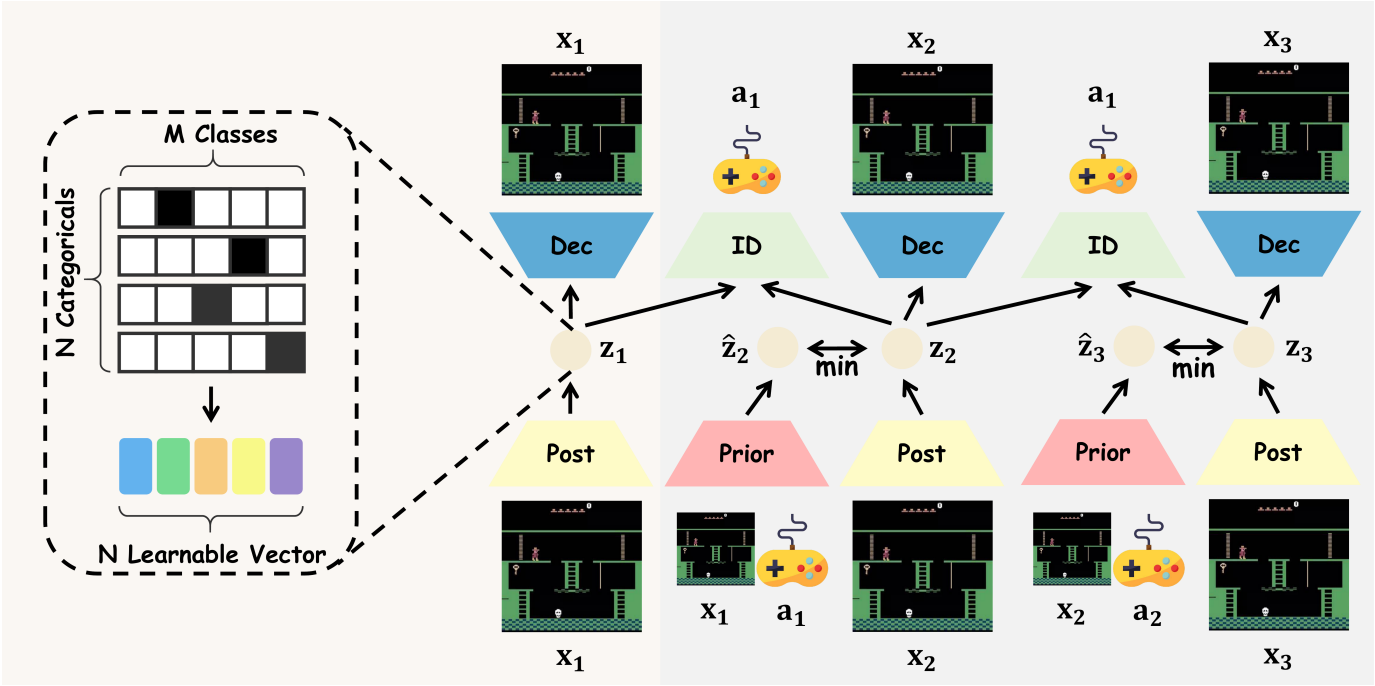



Fig. 2: Model Framework: Components include Posterior Model (Post), Prior Model (Prior), Inverse Dynamic Model (ID), and Decoder (Dec). We use reconstruction and inverse dynamics losses to learn effective latent distributions, minimizing the KL divergence between prior and posterior distributions.

### *B. Modeling State Representation*

The latent state [34], [35] $\mathcal{X}$ is represented by a **discrete vector distribution**, constructed by projecting categorical samples $\mathbf{c} \sim \mathrm{Cat}(K)$ onto learnable embeddings $\mathbf{E} \in \mathbb{R}^{K \times D}$, yielding continuous latent vectors $\mathbf{z} = \mathbf{E}^\top \mathbf{c}$.

We replace conventional Gaussian representations with discrete categorical distributions to better capture compositional structure (e.g., object combinations) [36], [37]. Through learnable embeddings $\mathbf{E}$, we transform one-hot categorical samples into dense continuous vectors $\mathbf{z}$, resolving sparsity issues while preserving topology. Straight-through gradient estimation enables effective backpropagation. The resulting latent vectors form a compressed, information-dense state space.

### *C. World Model Learning*

Our objective is to maximize the joint likelihood $\log p(x_{1:T}, a_{0:T-1} \mid x_0)$. Since this is intractable, we optimize the evidence lower bound (ELBO):

$$\begin{aligned}\mathcal{L} &= \log p(x_{1:T}, a_{0:T-1} | x_0) \\ &\geq \sum_{t=1}^{T} \Bigg[ \mathbb{E} \Bigg( \underbrace{\log p(x_t | z_t)}_{\text{Reconstruction loss}} + \underbrace{\log p(a_{t-1} | z_{t-1}, z_t)}_{\text{Inverse dynamic loss}} \Bigg) \\ &\quad - \underbrace{\mathrm{KL}\left(p(z_t | x_t) \parallel p(z_t | x_{t-1}, a_{t-1})\right)}_{\text{Forward dynamic loss}} \Bigg]\end{aligned} \tag{2}$$

where $z_t \sim p_\phi(z|x_t)$ and $z_{t-1} \sim p_\phi(z|x_{t-1})$ are posterior samples.

These terms have complementary roles: *Reconstruction loss* preserves information; *Inverse dynamic loss* captures action-latent correlations; *Forward dynamic loss* regularizes the posterior $p(z_t|x_t)$ to match the prior $p(z_t|x_{t-1}, a_{t-1})$, enabling predictive dynamics.

Our world model comprises four components: posterior model, prior model, inverse dynamics model, and decoder (Figure 2).

$$\begin{cases} \text{Posterior Model} & z_t \sim p_\phi(z|x_t) \\ \text{Prior Model} & \hat{z}_t \sim p_\theta(z|x_{t-1}, a_{t-1}) \\ \text{Inverse Dynamic Model} & a_{t-1} \sim p_\psi(a|z_{t-1}, z_t) \\ \text{Decoder} & \hat{x}_t \sim p_\delta(x|z_t) \end{cases}$$

All components are implemented as neural networks, with $\phi, \theta, \psi, \delta$ representing their combined parameter vector. The inverse dynamics model and decoder encourage the posterior model to learn a latent distribution that is relevant to the environment, while the prior model approximates the posterior model to enhance predictive capability.

The KL term requires special treatment as simultaneously training $p_\theta$ while regularizing $q_\phi$ against an immature prior risks posterior collapse. We introduce KL balancing via asymmetric learning rates:

$$\eta_\theta \nabla_\theta \mathrm{KL}(q_\phi \parallel p_\theta) - \eta_\phi \nabla_\phi \mathrm{KL}(q_\phi \parallel p_\theta) \tag{3}$$

with $\eta_\theta > \eta_\phi$ prioritizing prior training. This isolates representational learning from dynamical inaccuracies during early training phases.

| | GME(ours) | ICM | RIDE | E3B |
|---|---|---|---|---|
| BeamRider | **5336.21**$_{\pm 193.94}$ | 4685.98$_{\pm 297.56}$ | 4524.99$_{\pm 308.35}$ | 4977.10$_{\pm 434.87}$ |
| Berzerk | **3776.66**$_{\pm 107.52}$ | 2830.06$_{\pm 309.50}$ | 2798.26$_{\pm 549.36}$ | 2619.30$_{\pm 485.65}$ |
| Centipede | **10631.72**$_{\pm 525.77}$ | 9187.88$_{\pm 188.40}$ | 8551.51$_{\pm 339.54}$ | 9260.11$_{\pm 1139.46}$ |
| Freeway | **22.84**$_{\pm 1.75}$ | 20.72$_{\pm 2.59}$ | 20.77$_{\pm 0.55}$ | 21.81$_{\pm 1.41}$ |
| MontezumaRevenge | **2345.40**$_{\pm 78.69}$ | 1462.00$_{\pm 738.44}$ | 400.20$_{\pm 0.40}$ | 1266.00$_{\pm 1007.11}$ |
| Bowling | **38.15**$_{\pm 5.32}$ | 30.81$_{\pm 0.66}$ | 30.74$_{\pm 0.77}$ | 31.06$_{\pm 1.42}$ |
| Gravitar | **1277.60**$_{\pm 118.43}$ | 927.90$_{\pm 55.47}$ | 843.10$_{\pm 49.77}$ | 1023.90$_{\pm 265.16}$ |
| Skiing | **-17120.21**$_{\pm 10529.40}$ | $-30000.00_{\pm 0.00}$ | $-25802.60_{\pm 8394.80}$ | $-30646.22_{\pm 1292.44}$ |
| Venture | **1024.00**$_{\pm 53.21}$ | 885.00$_{\pm 90.09}$ | 895.60$_{\pm 70.67}$ | 926.80$_{\pm 47.73}$ |
| Basic | **10.29**$_{\pm 0.08}$ | 10.16$_{\pm 0.10}$ | 10.20$_{\pm 0.15}$ | 10.21$_{\pm 0.08}$ |
| Corridor | **196.45**$_{\pm 52.10}$ | 124.38$_{\pm 12.12}$ | 139.67$_{\pm 32.38}$ | 160.25$_{\pm 17.69}$ |
| DefendCenter | **160.03**$_{\pm 25.20}$ | 137.70$_{\pm 7.07}$ | 139.60$_{\pm 24.47}$ | 143.08$_{\pm 14.35}$ |
| DeathMatch | **70.55**$_{\pm 8.38}$ | 66.98$_{\pm 8.29}$ | 64.62$_{\pm 4.99}$ | 63.88$_{\pm 9.00}$ |
| PredictPosition | **0.42**$_{\pm 0.06}$ | 0.24$_{\pm 0.14}$ | 0.18$_{\pm 0.12}$ | 0.30$_{\pm 0.14}$ |
| TakeCover | **357.93**$_{\pm 36.51}$ | 331.27$_{\pm 34.69}$ | 341.03$_{\pm 46.23}$ | 353.04$_{\pm 32.06}$ |
| SuperMarioBros-2-4-v3 | **21.11**$_{\pm 0.24}$ | 20.84$_{\pm 0.31}$ | 20.87$_{\pm 0.22}$ | 21.04$_{\pm 0.26}$ |
| SuperMarioBros-4-4-v3 | **148.39**$_{\pm 19.72}$ | 119.65$_{\pm 66.75}$ | 136.06$_{\pm 29.54}$ | 101.30$_{\pm 56.08}$ |
| SuperMarioBros-8-4-v3 | **141.61**$_{\pm 12.56}$ | 123.24$_{\pm 61.30}$ | 114.46$_{\pm 18.94}$ | 114.74$_{\pm 15.06}$ |
| WalkerStand | **879.10**$_{\pm 64.72}$ | 857.01$_{\pm 92.94}$ | 849.89$_{\pm 69.19}$ | 819.65$_{\pm 93.94}$ |
| WalkerWalk | **734.44**$_{\pm 117.20}$ | 708.50$_{\pm 100.01}$ | 724.31$_{\pm 116.25}$ | 686.84$_{\pm 160.91}$ |
| WalkerRun | **251.70**$_{\pm 40.90}$ | 217.91$_{\pm 69.45}$ | 244.13$_{\pm 31.19}$ | 245.35$_{\pm 41.17}$ |
| HopperHop | **147.65**$_{\pm 16.02}$ | 108.51$_{\pm 58.51}$ | 112.87$_{\pm 41.74}$ | 111.70$_{\pm 41.43}$ |
| HopperStand | **522.43**$_{\pm 189.19}$ | 444.68$_{\pm 245.13}$ | 476.49$_{\pm 182.09}$ | 498.30$_{\pm 239.06}$ |
| CheetahRun | **441.03**$_{\pm 43.75}$ | 437.97$_{\pm 40.67}$ | 427.66$_{\pm 31.19}$ | 434.44$_{\pm 31.95}$ |

TABLE I: Results on 9 Atari games, 6 Vizdoom games, 3 Mario games, and 6 DeepMind Control environments with different exploration difficulties. We ran 10 million steps on each environment for Atari, 1 million steps for Vizdoom, 10 million steps for Mario, and 5 million steps for DeepMind Control. We reported the mean and standard deviation of returns over 5 seeds, with the highest return values highlighted in bold. Our algorithm achieved the highest rewards across all environments.

### D. World Model for Exploration

We derive an information-theoretic intrinsic reward from our world model. The mutual information $I(z_t; x_t \mid h_t)$ between latent state $z_t$ and observation $x_t$, given history $h_t$, quantifies the *information gain* when transitioning from prior belief $p_\theta(z_t \mid h_t)$ to posterior $p_\phi(z_t \mid x_t)$:

$$\begin{aligned} I(z_t; x_t \mid h_t) &= \mathbb{E}_{p(z_t, x_t \mid h_t)} \left[ \log \frac{p(z_t, x_t \mid h_t)}{p(z_t \mid h_t) p(x_t \mid h_t)} \right] \\ &= \mathbb{E}_{x_t \sim p(x_t \mid h_t)} \mathbb{E}_{z_t \sim p(z_t \mid x_t, h_t)} \left[ \log \frac{p(z_t \mid x_t, h_t)}{p(z_t \mid h_t)} \right] \\ &= \mathbb{E}_{x_t \sim p(\cdot \mid h_t)} \left[ \mathrm{KL} \left( p(z_t \mid x_t, h_t) \parallel p(z_t \mid h_t) \right) \right] \end{aligned} \tag{4}$$

Simultaneously, we capture *epistemic uncertainty* via conditional entropy $\mathcal{H}(z_t \mid h_t)$. Combining these yields our exploration reward:

$$r_t^{int} = \underbrace{\mathcal{H}\big(p_\theta(z_t \mid h_t)\big)}_{\text{Epistemic Uncertainty}} + \underbrace{\mathrm{KL}\left(p_\phi(z_t \mid x_t) \parallel p_\theta(z_t \mid h_t)\right)}_{\text{Information Gain}} \tag{5}$$

This formulation offers three key advantages: (1) Discrete latents allow exact entropy computation; (2) The probabilistic transition operator explicitly quantifies visitation uncertainty; (3) The KL term measures Bayesian surprise directly. Crucially, this incentivizes exploration towards *high-entropy transitions* and *high-information states*.

## V. Experiments

### A. Environments

We evaluate our algorithm across diverse high-dimensional visual domains using four suites: **Atari 2600** (ALE) with nine games of varying exploration difficulty (e.g., *BeamRider, MontezumaRevenge, Venture*); **ViZDoom** with six timed first-person scenarios (e.g., *Basic, Corridor, TakeCover*); **Super Mario Bros** with three progressively harder levels (*SuperMarioBros-2-4-v3, -4-4-v3, -8-4-v3*); and the visual variants of the **DeepMind Control Suite** featuring six locomotion tasks (*WalkerStand/Walk/Run, HopperHop/Stand, CheetahRun*). This mix spans 1st/3rd-person views, discrete/continuous actions, fixed goals/locomotion control, and varying reward sparsity, enabling robust assessment of exploration. For comparison, we include state-of-the-art exploration baselines: **ICM** (intrinsic curiosity) [13], **RIDE** (impact-driven intrinsic rewards) [15], and **E3B** (elliptical episodic bonuses) [12] .

### B. Baseline

We selected several state-of-the-art exploration algorithms for comparison, representing different categories: **ICM** [13] uses the Intrinsic Curiosity Module to guide the model's exploration; **RIDE** [15] employs Rewarding Impact-Driven Exploration to model intrinsic rewards and **E3B** designs unique Elliptical Episodic Bonuses to replace pseudo-counts.

### C. Main Results of Exploration Efficiency Experiment

Table I shows that GME outperforms baselines across all 24 environments. In Atari's exploration-intensive games and

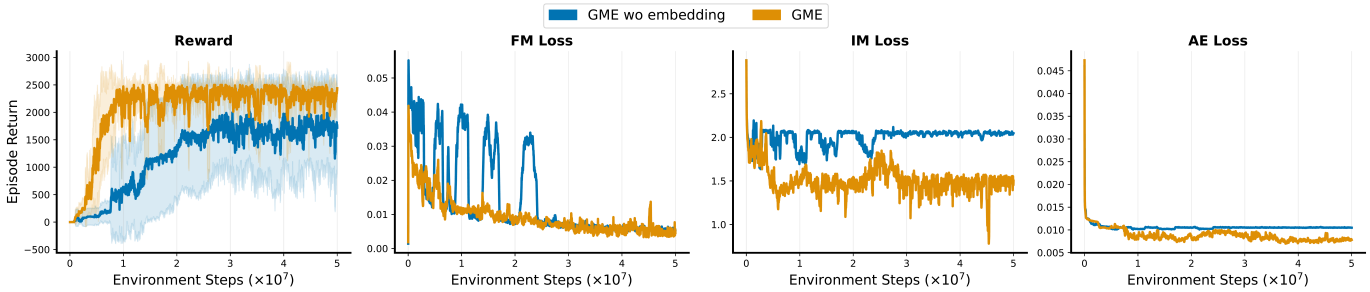
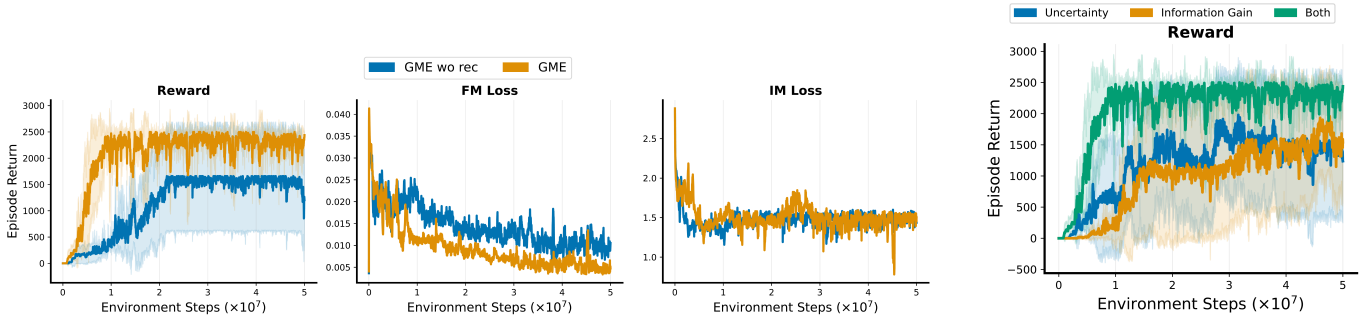



Fig. 3: Ablation for Learnable Embeddings. We designed comparative experiments to illustrate the effectiveness of learnable embeddings. We report the algorithm's performance and the loss curves during training.




(a) Ablation for reconstruction loss

(b) Ablation for reward shaping

Fig. 4: We designed comparative experiments to demonstrate the effectiveness of reconstruction loss and our reward design. We compared different variants of GME and reported the algorithm's loss curves and performance.

ViZDoom's first-person tasks, GME achieves the highest sample efficiency. It also dominates in Super Mario Bros and DeepMind Control tasks, validating its robustness.

Notably, GME delivers the most remarkable improvements in challenging tasks: in Montezuma's Revenge, it outperforms ICM by 60.4% (2345.40 vs 1462.00); in Vizdoom's Predict-Position, it achieves a 40.0% enhancement over E3B (0.42 vs 0.30); and in Mario's 8-4 level, it gains 15.0% over ICM. In continuous control, GME outperforms RIDE by 30.8% in HopperHop. These results demonstrate GME's high sample efficiency in sparse-reward settings.

### *D. Further Analysis*

*1) Generalize to New Environments:* In this section, we assess GME's generalization on novel environments using the **Procgen** benchmark. We pre-train GME on unlimited simple levels before training the agent on randomly sampled new levels, keeping world model parameters frozen. We compare DrAC versus DrAC+GME.

As shown in Table II, incorporating GME yields higher rewards and sample efficiency, indicating our world model's ability to generalize to unseen environments.

*2) Generalize to New Tasks:* We evaluate generalization on novel tasks using DeepMind Control. We pre-train world models on 'stand' tasks for Walker and Hopper robots, then apply them (frozen) to new tasks using PPO.

| Environment | DrAC+GME(ours) | DrAC |
|---|---|---|
| chaser | **9.28**$_{\pm 4.44}$ | 5.94$_{\pm 4.77}$ |
| climber | **13.24**$_{\pm 0.91}$ | 2.65$_{\pm 0.19}$ |
| fruitbot | **3.58**$_{\pm 1.67}$ | 2.03$_{\pm 0.78}$ |
| miner | **5.38**$_{\pm 5.40}$ | 1.24$_{\pm 0.58}$ |
| plunder | **5.77**$_{\pm 1.04}$ | 2.71$_{\pm 1.68}$ |
| starpilot | **25.90**$_{\pm 8.39}$ | 12.65$_{\pm 10.96}$ |

TABLE II: Experimental Results for Generalization to New Environments: we pre-trained GME on infinite simple difficulty levels and tested it on three randomly sampled levels.

| Task | PPO+GME(ours) | PPO |
|---|---|---|
| HopperHop | **134.80**$_{\pm 29.95}$ | 112.39$_{\pm 29.98}$ |
| WalkerWalk | **669.80**$_{\pm 78.23}$ | 572.08$_{\pm 68.64}$ |
| WalkerRun | **260.94**$_{\pm 27.60}$ | 219.69$_{\pm 48.05}$ |

TABLE III: Generalization to New Tasks: Pretrained world models applied to distinct tasks.

Table III shows that GME enhances performance, demonstrating that learned dynamics facilitate adaptation to new tasks.

*3) Ablation for Learnable Embeddings:* To validate our learnable embeddings, we compared GME with a variant lacking them. Figure 3 shows significant performance decline without embeddings. The loss curves reveal that embeddings enhance latent distribution expressivity, reducing both autoencoder and inverse dynamics losses and stabilizing forward

model learning.

*4) Ablation for Reconstruction Loss:* We introduced reconstruction loss to encourage learning environmental details beyond agent dynamics. Figure 4a shows that removing this loss significantly degrades performance. While inverse dynamics remains stable, reconstruction loss is crucial for learning richer latent features.

*5) Ablation for Reward Shaping:* We compared GME with variants using only information gain or epistemic uncertainty. Figure 4b confirms that combining both yields the best results. Relying solely on one measure limits exploration efficiency, whereas their combination ensures a balance between exploring stochastic transitions and informative states.

## VI. Conclusion

We addressed representation limitations by introducing Stochastic Latent-MDP and a stochastic state representation. We constructed a world model to compute intrinsic rewards suitable for high-dimensional inputs. Experiments across Atari, ViZDoom, Super Mario Bros, and DeepMind Control demonstrate that GME significantly outperforms baselines, enhancing sample efficiency. Ablation studies validate our design choices.

## References


[1] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*, 2016, pp. 1928–1937.

[2] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, p. 484, 2016.

[3] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, p. 354, 2017.

[4] M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, "Unifying count-based exploration and intrinsic motivation," in *Advances in Neural Information Processing Systems*, 2016, pp. 1471–1479.

[5] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "Exploration by random network distillation," *arXiv preprint arXiv:1810.12894*, 2018.

[6] A. Ecoffet, J. Huizinga, J. Lehman, K. O. Stanley, and J. Clune, "Go-explore: a new approach for hard-exploration problems," *arXiv preprint arXiv:1901.10995*, 2019.

[7] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, and A. A. Efros, "Large-scale study of curiosity-driven learning," *arXiv preprint arXiv:1808.04355*, 2018.

[8] A. L. Strehl and M. L. Littman, "An analysis of model-based interval estimation for markov decision processes," *Journal of Computer and System Sciences*, vol. 74, no. 8, pp. 1309–1331, 2008.

[9] H. Tang, R. Houthooft, D. Foote, A. Stooke, O. X. Chen, Y. Duan, J. Schulman, F. DeTurck, and P. Abbeel, "# exploration: A study of count-based exploration for deep reinforcement learning," in *Advances in neural information processing systems*, 2017, pp. 2753–2762.

[10] M. C. Machado, M. G. Bellemare, and M. Bowling, "Count-based exploration with the successor representation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5125–5133.

[11] D. Jo, S. Kim, D. Nam, T. Kwon, S. Rho, J. Kim, and D. Lee, "Leco: Learnable episodic count for task-specific intrinsic reward," *Advances in Neural Information Processing Systems*, vol. 35, pp. 30 432–30 445, 2022.

[12] M. Henaff, R. Raileanu, M. Jiang, and T. Rocktäschel, "Exploration via elliptical episodic bonuses," *Advances in Neural Information Processing Systems*, vol. 35, pp. 37 631–37 646, 2022.

[13] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *International conference on machine learning*. PMLR, 2017, pp. 2778–2787.

[14] D. Pathak, D. Gandhi, and A. Gupta, "Self-supervised exploration via disagreement," in *International conference on machine learning*. PMLR, 2019, pp. 5062–5071.

[15] R. Raileanu and T. Rocktäschel, "Ride: Rewarding impact-driven exploration for procedurally-generated environments," *arXiv preprint arXiv:2002.12292*, 2020.

[16] M. Mutti, L. Pratissoli, and M. Restelli, "A policy gradient method for task-agnostic exploration," in *4th Lifelong Machine Learning Workshop at ICML 2020*, 2020.

[17] Y. Seo, L. Chen, J. Shin, H. Lee, P. Abbeel, and K. Lee, "State entropy maximization with random encoders for efficient exploration," in *International Conference on Machine Learning*. PMLR, 2021, pp. 9443–9454.

[18] P. Wu, A. Escontrela, D. Hafner, P. Abbeel, and K. Goldberg, "Daydreamer: World models for physical robot learning," in *Conference on robot learning*. PMLR, 2023, pp. 2226–2240.

[19] Y. Shang, Y. Lin, Y. Zheng, H. Fan, J. Ding, J. Feng, J. Chen, L. Tian, and Y. Li, "Urbanworld: An urban world model for 3d city generation," *arXiv preprint arXiv:2407.11965*, 2024.

[20] Z. Zhu, X. Wang, W. Zhao, C. Min, N. Deng, M. Dou, Y. Wang, B. Shi, K. Wang, C. Zhang *et al.*, "Is sora a world simulator? a comprehensive survey on general world models and beyond," *arXiv preprint arXiv:2405.03520*, 2024.

[21] J. Ding, Y. Zhang, Y. Shang, Y. Zhang, Z. Zong, J. Feng, Y. Yuan, H. Su, N. Li, N. Sukiennik *et al.*, "Understanding world or predicting future? a comprehensive survey of world models," *arXiv preprint arXiv:2411.14499*, 2024.

[22] N. Hansen, H. Su, and X. Wang, "Td-mpc2: Scalable, robust world models for continuous control," *arXiv preprint arXiv:2310.16828*, 2023.

[23] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba, "Mastering atari with discrete world models," *arXiv preprint arXiv:2010.02193*, 2020.

[24] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering diverse domains through world models," *arXiv preprint arXiv:2301.04104*, 2023.

[25] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[26] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. De Turck, and P. Abbeel, "Vime: Variational information maximizing exploration," *Advances in neural information processing systems*, vol. 29, 2016.

[27] N. Haber, D. Mrowca, S. Wang, L. F. Fei-Fei, and D. L. Yamins, "Learning to play with intrinsically-motivated, self-aware agents," *Advances in neural information processing systems*, vol. 31, 2018.

[28] S. Lobel, A. Bagaria, and G. Konidaris, "Flipping coins to estimate pseudocounts for exploration in reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2023, pp. 22 594–22 613.

[29] M. Yuan, R. C. Castanyer, B. Li, X. Jin, G. Berseth, and W. Zeng, "Rlexplore: Accelerating research in intrinsically-motivated reinforcement learning," *arXiv preprint arXiv:2405.19548*, 2024.

[30] X. Yu, Y. Lyu, and I. Tsang, "Intrinsic reward driven imitation learning via generative model," in *International conference on machine learning*. PMLR, 2020, pp. 10 925–10 935.

[31] M. Yuan, B. Li, X. Jin, and W. Zeng, "Automatic intrinsic reward shaping for exploration in deep reinforcement learning," in *International Conference on Machine Learning*. PMLR, 2023, pp. 40 531–40 554.

[32] A. Aubret, L. Matignon, and S. Hassas, "An information-theoretic perspective on intrinsic motivation in reinforcement learning: A survey," *Entropy*, vol. 25, no. 2, p. 327, 2023.

[33] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[34] M. Watter, J. Springenberg, J. Boedecker, and M. Riedmiller, "Embed to control: A locally linear latent dynamics model for control from raw images," *Advances in neural information processing systems*, vol. 28, 2015.

[35] D. Ha and J. Schmidhuber, "World models," *arXiv preprint arXiv:1803.10122*, 2018.

[36] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 873–12 883.

[37] A. Razavi, A. Van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," *Advances in neural information processing systems*, vol. 32, 2019.