applied sciences

Article

Balanced Domain Randomization for Safe Reinforcement Learning

Cheongwoong Kang [1], Wonjoon Chang [1] and Jaesik Choi [1,2,*]

[1] Kim Jaechul Graduate School of AI, Korea Advanced Institute of Science and Technology, Daejeon 34141, Republic of Korea; cw.kang@kaist.ac.kr (C.K.); one_jj@kaist.ac.kr (W.C.)
[2] INEEJI Corporation, Seongnam 13558, Republic of Korea
* Correspondence: jaesik.choi@kaist.ac.kr

Abstract: Reinforcement Learning (RL) has enabled autonomous agents to achieve superhuman performance in diverse domains, including games, navigation, and robotic control. Despite these successes, RL agents often struggle with overfitting to specific training environments, which can hinder their adaptability to new or rare scenarios. Domain randomization is a widely used technique designed to enhance policy robustness by training models in a variety of randomized contexts. In traditional domain randomization, however, models tend to prioritize learning from more common domains, often neglecting rare ones. To address this imbalance, we propose Balanced Domain Randomization (BDR) that balances the learning focus based on the rarity of contexts. We assess context rarity in the embedding space using statistical analysis of context vectors and adjust the loss weights for each transition based on this rarity. This ensures that the agent dedicates adequate attention to rare domains. Our experimental results show that BDR efficiently enhances worst-case performance, significantly improving the robustness of RL-based robotic controllers across diverse conditions. This study provides a robust framework for RL agents, reducing risks in uncommon scenarios and ensuring reliable performance in varied environments.

Keywords: reinforcement learning; domain randomization; generalization; robustness; safety

check for updates

Citation: Kang, C.; Chang, W.; Choi, J. Balanced Domain Randomization for Safe Reinforcement Learning. *Appl. Sci.* **2024**, *14*, 9710. https://doi.org/10.3390/app14219710

Academic Editor: Antonio Pagliaro

Received: 26 September 2024
Revised: 18 October 2024
Accepted: 21 October 2024
Published: 24 October 2024

1. Introduction

Reinforcement Learning (RL) has significantly advanced the capabilities of autonomous agents to perform complex tasks across various domains [1–3], including board games [4,5], real-time strategy games [6,7], robotic manipulation [8,9], and navigation [10,11]. The core strength of RL lies in its ability to autonomously learn policies from environmental interactions through trial and error. However, RL models often overfit to the specific contexts encountered during training, severely limiting their adaptability and robustness in real-world applications where scenarios may differ significantly from their training data.

Domain randomization is a widely adopted technique to enhance the robustness of RL agents by exposing them to various simulation environments during training [9,12–17]. Despite its success, models often prioritize learning from frequent domains with similar dynamics, resulting in insufficient learning from rare but potentially critical scenarios. This imbalance can result in suboptimal policies that perform well in common situations, but fail in less frequent, challenging ones. To develop truly robust RL agents, it is essential to ensure adequate training on these rare domains.

In this paper, we propose a novel approach that balances the learning focus based on the rarity of contexts (as illustrated in Figure 1). We measure the rarity of a context in the embedding space using the statistics of context vectors, under an assumption that policies may not be sufficiently trained in contexts where their embeddings significantly deviate from the distribution. We adjust the training loss weights according to the rarity of these contexts, encouraging the agent to focus more on rare ones. This method ensures that the agent is thoroughly exposed to a wide range of domains. Our main contributions are as follows:

- Identification of rare domains: we identify rare domains where policies may be undertrained by analyzing context embedding distances.
- Balancing domain randomization: we propose a balancing mechanism that assigns greater weight to rare domains during training to achieve balanced domain randomization.
- Empirical validation: our experiments demonstrate that the proposed method efficiently improves worst-case performance, enhancing the robustness of RL agents, especially in novel situations.

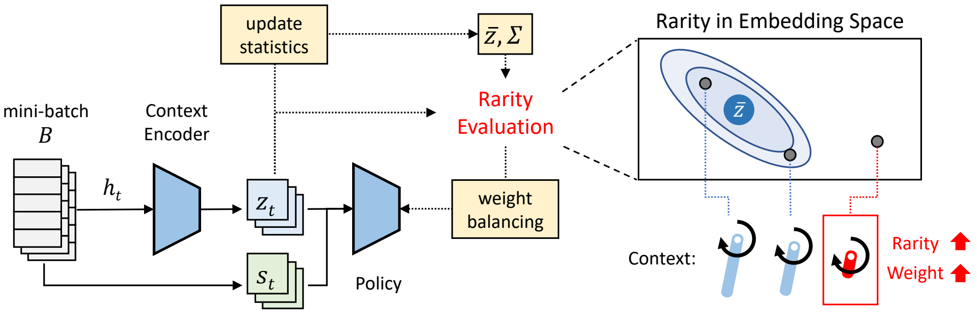


Figure 1. An overall framework of Balanced Domain Randomization (BDR). BDR reweights training domains based on their rarity in the embedding space. The framework uses a context encoder to capture contextual information about the current environment from a recent history of state–action pairs. The rarity evaluation module estimates the rarity of each context, adaptively adjusting the weight assigned to each domain. This ensures that rare scenarios are sufficiently visited during training.

In summary, this study addresses the critical need for robustness in RL by introducing a balanced domain randomization approach that ensures a harmonious representation of both frequent and rare domains, improving the reliability of RL agents for deployment in diverse and unpredictable environments.

The remainder of this paper is organized as follows: Section 2 reviews related work on contextual RL and domain randomization. Section 3 explains the contextual reinforcement learning framework, and discusses the imbalance problem in standard domain randomization. In Section 4, we propose the Balanced Domain Randomization (BDR) method, detailing our training framework and context rarity evaluation based on the statistics of context embedding. Section 5 shows the experimental results on benchmark tasks. We conclude with a discussion of our findings and future directions in Section 6.

2. Related Work

Contextual Reinforcement Learning (RL) extends the standard RL framework by incorporating environmental contexts, which represent varying physical properties that affect the transition dynamics of environments. This approach enables agents to learn policies that generalize across a range of different situations. Contextual RL has been extensively studied to develop adaptable and robust RL agents, with applications ranging from game playing to robotic control, where agents must adapt to different levels or environmental dynamics [18,19].

Domain randomization is a prominent technique employed to improve the robustness and generalization of RL agents by exposing them to a diverse set of simulated environments during training [9,12–17,20–23]. By varying the environmental contexts, such as physical properties, domain randomization encourages the agent to learn policies that are adaptable across various environments. This approach has been particularly successful in sim-to-real transfer, where agents trained in simulation are deployed in the real world [24–26]. However, while domain randomization improves generalization, RL agents often prioritize learning in certain domains, which may share similar dynamics. Our analysis shows that uniformly sampling domains does not ensure balanced learning, leaving rare but critical scenarios underrepresented. This imbalance within domain randomization is the core challenge addressed by our work.

Data imbalance is a well-documented issue across various machine learning domains. To mitigate its effects, several approaches have been proposed. Data rebalancing techniques, such as oversampling rare classes or undersampling common ones, aim to create more balanced datasets [27–29]. Another strategy is loss reweighting, which adjusts the training weights to emphasize rare classes, thus generating more balanced training signals [30–34]. Transfer learning approaches have also been used to transfer features learned from common classes to rare ones [35,36]. Additionally, curriculum learning methods gradually introduce more complex or rarer scenarios as training progresses, encouraging the agent to learn simpler tasks before more challenging ones [37–39].

Our proposed method builds on the concept of loss reweighting, but addresses an additional challenge: autonomously detecting rare domains. Existing approaches in visual domains have attempted to measure the rarity of high-dimensional data using embedding distances [40,41]. Inspired by these approaches, we evaluate the rarity of contexts based on their embedding vectors in contextual RL. Our proposed method effectively mitigates data imbalance by integrating loss reweighting with the rarity measure, ensuring that RL agents receive adequate exposure to rare scenarios.

3. Problem Definition

3.1. Contextual Reinforcement Learning Framework

We introduce a contextual Reinforcement Learning (RL) framework for training agents under varying contextual conditions, where the context encapsulates the physical properties of the environment. This framework is grounded in the contextual Markov Decision Process (MDP) [18], defined as $M = (S, A, T, R, \gamma; c)$. In this formulation, S represents the set of states, A denotes the set of actions, and $T(s'|s, a; c)$ is the transition function, indicating the probability of moving from state s to s', given action a in the context c. The reward function $R(s, a, s'; c)$ provides the immediate reward given the transition, and the discount factor γ models the trade-off between current and future rewards. The context c defines system parameters that determine the physical properties of the environment. In domain randomization, environments are considered to belong to the same domain when they share the same context or system parameters.

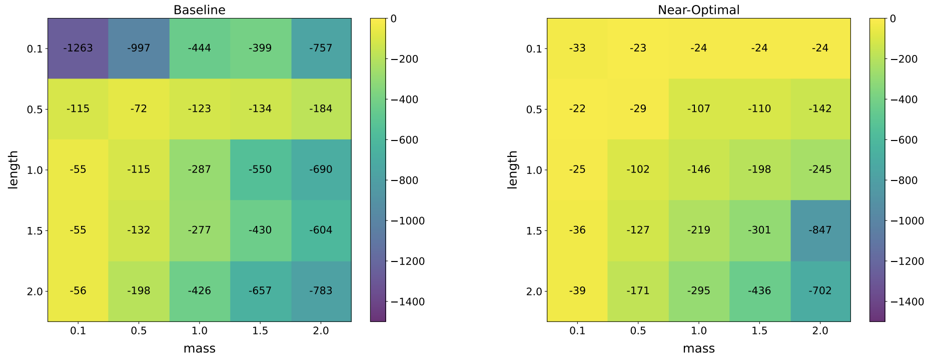
In practical scenarios, the context c may not be directly observable, making the contextual MDP a special case of Partially Observable MDPs (POMDPs) [42,43], where contexts act as hidden variables. Under this partial observability, contextual RL has two main objectives: (1) encoding a history of recent transitions into a context embedding, and (2) learning a contextual policy $\pi(s; c) : S \times C \to A$ that determines the optimal action to maximize the expected return $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t, s_{t+1}; c_t)\right]$. Here, the context encoder is utilized to produce a context embedding using neural networks [14]. Specifically, it maps the recent state–action history into a context embedding, which implicitly estimates the underlying environmental contexts. Then, the context embedding is used as an additional input of the policy for more informed decision-making.

3.2. Learning Imbalance in Domain Randomization

Domain randomization enhances the adaptability of RL agents by training them across various environmental contexts. However, our analysis reveals a significant limitation: RL agents often prioritize learning in specific domains, which can lead to suboptimal performance in others.

To explore this issue, we conduct experiments on the Pendulum task, where mass and length are randomized as system parameters. We compare the performance of a baseline model trained with Uniform Domain randomization (UDR) to near-optimal policies trained on individual domains. Figure 2 presents the average episodic return of the models across different domains. The baseline model shows inconsistent performance across various contexts, with notable deficits when the pendulum length is 0.1. In contrast, the results of the near-optimal policies, which represent the potential upper bounds of performance, demonstrate that a policy can be sufficiently trained, even when the pendulum length is 0.1.

These results suggest that RL agents, when trained on uniformly sampled domains, tend to prioritize specific domains over others. We hypothesize that this imbalance stems from the relative rarity of domains in the context embedding space. Although this imbalance might enhance the average performance in common situations, it undermines the agent's adaptability in rare but potentially critical scenarios, presenting significant challenges in real-world applications.



(**a**) Baseline (**b**) Near-optimal solution

Figure 2. Episodic returns of the baseline and near-optimal policies in the pendulum task. The number in a cell indicates the average episodic return in each domain. The baseline policy (**a**) exhibits lower returns when the pendulum length is 0.1 compared to the near-optimal policies (**b**), demonstrating that policies trained under domain randomization tend to prioritize specific domains over others even though domains are uniformly sampled.

4. Balanced Domain Randomization

To address the imbalance inherent in domain randomization, we propose a method that balances the learning proportions of contexts according to their rarity in the embedding space. We measure the rarity of a context based on the distance between its context vector and the center of context embeddings. This measure of rarity is then used to evaluate the importance of each context during training, ensuring that the agent pays sufficient attention to rare domains. Our approach aims to improve the agent's performance in worst-case scenarios, enhancing overall robustness and reliability. The overall procedure is illustrated in Figure 1 and detailed in Algorithm 1.

Algorithm 1: Balanced Domain Randomization

```
Initialize context encoder $f$ and RL network $\theta$;
Initialize average context vector $\bar{z} \leftarrow \mathbf{0}$;
Initialize covariance matrix $\Sigma \leftarrow \mathbf{I}$;
/* Learning Phase                                              */
for each update step $j$ do
    Sample a random mini-batch of transitions $B_j = (S_j, A_j, R_j, S_{j+1}, H_j)$ from
      replay buffer $\mathcal{D}$
    Compute the mean of context embeddings in the mini-batch:
      $\bar{f}(H_j) = \sum_{h_t \in H_j} f(h_t) / |H_j|$
    Update the statistics $\bar{z}_j, \Sigma_j$
    Compute rarity scores: $\mathrm{Rarity}(h_t) \leftarrow d(f(h_t), \bar{z}_j\,;\,\Sigma_j)$ for each $h_t$ in $H_j$
    Assign weights using softmax:
      $w(h_t) \leftarrow \exp(\mathrm{Rarity}(h_t)) / \sum_{h_t \in H_j} \exp(\mathrm{Rarity}(h_t))$
    Compute weighted RL loss: $L_{\mathrm{weighted}}(B_j) = \sum_{\tau_t \in B_j} \{\beta w(h_t) + (1-\beta)\}\, l_{\mathrm{RL}}(\tau_t)$
    Update RL network using weighted RL loss
```

4.1. Context Embedding

In our training framework, the agent interacts with the environment to collect data stored in a replay buffer, which is used for learning steps. At the j-th learning step, we sample a mini-batch B_j of transitions from the replay buffer. $\tau_t \in B_j$ is a transition of time t stored in the buffer. τ_t includes the current state s_t, the action a_t, the reward r_t, the next state s_{t+1}, and the history of state–action pairs h_t. The context encoder f maps h_t into a context vector z_t, which is then fed into the policy as a condition. The context vector captures contextual information from the agent's past interactions with the environment. This structure allows the agent to take adaptive actions based on the current environment dynamics, even when it gains the same state as its input.

4.2. Accessing the Rarity of Contexts

As explained in Section 3.2, RL agents tend to prioritize specific domains over others in randomized environments. We hypothesize that this imbalance problem in domain randomization arises from the rarity of contexts encountered during training. It is caused by the typical learning framework that the policy is trained to increase the expected reward of its rollouts, which may lead to overfitting to domains with common dynamics. In this regard, it is necessary to evaluate the rarity of the contexts to detect domains in which the policy may potentially underfit.

Given a history h_t, the corresponding context vector is computed as $z_t = f(h_t)$. We define the distance between z_t and the center of context embeddings as its rarity,

$$\text{Rarity}(h_t) = d(f(h_t), \bar{z}) \tag{1}$$

where $d(\cdot, \cdot)$ is a distance metric and $\bar{z}$ represents the expected context vector. Any distance metric can be employed, such as Euclidean or cosine distance.

Without any specific restrictions to context vectors, variables in the context embedding usually have different variances and exhibit correlations with each other. It is not considered in traditional distance measures like Euclidean distance, since it assumes all features are independent and equally scaled. Therefore, we utilize the Mahalanobis distance [44,45], which accounts for both the variance of individual variables and the correlations between them using the covariance matrix. This makes it particularly well-suited for high-dimensional spaces, such as our context embedding space, where features often exhibit complex relationships. Mathematically, the Mahalanobis distance between the embedding vector z_t and the distribution with mean $\bar{z}$ and covariance matrix Σ is given by

$$d(f(h_t), \bar{z}\,;\, \Sigma) = \sqrt{(z_t - \bar{z})^\top \Sigma^{-1} (z_t - \bar{z})}. \tag{2}$$

This distance normalizes the difference between z_t and the mean $\bar{z}$ by the variance and correlations present in the embedding space, ensuring that deviations in directions with low variance are weighted more heavily.Consequently, Mahalanobis distance allows us to effectively capture rare contexts, which may be overlooked in traditional distance metrics.

4.3. Reweighting Training Loss

To ensure that the policy is adequately trained on rare domains during training, we adjust the training loss weights based on the computed rarity of each context. We introduce Balanced Domain Randomization (BDR), a method that encourages the agent to focus on transitions sampled from rare domains during updates, enhancing generalization performance. This approach involves the process of assigning weights to training samples by evaluating the rarity scores based on their context vectors. Specifically, we normalize the rarity scores to determine weights using the softmax function,

$$w(h_t) = \frac{\exp(\text{Rarity}(h_t))}{\sum_{h_t \in H_j} \exp(\text{Rarity}(h_t))} \tag{3}$$

where H_j is the set of histories in the j-th mini-batch. The weighted loss for the RL model is then computed as

$$L_{\text{weighted}}(B_j) = \sum_{\tau_t \in B_j} \beta\, w(h_t) l_{\text{RL}}(\tau_t) + (1-\beta)\, l_{\text{RL}}(\tau_t) = \sum_{\tau_t \in B_j} \{\beta w(h_t) + (1-\beta)\}\, l_{\text{RL}}(\tau_t) \tag{4}$$

where $l_{RL}(\tau_t)$ is the original RL loss for a single transition τ_t, $L_{\text{weighted}}(B_j)$ is the total weighted loss for a batch B_j, and β is a scalar value that balances the weighted loss and the original loss. In this work, β is set to 0.5.

4.4. Context Statistics with Exponential Moving Average

To compute the rarity of each context vector, we estimate the statistics of the context vectors. However, since the context encoder is continually updated during training, re-encoding all past histories in the replay buffer would be necessary to accurately compute these statistics. To reduce this computational burden, we employ Exponential Moving Average (EMA) to gradually update the approximation with mini-batch samples. The mean context vector is updated as follows:

$$\bar{z}_j = \alpha\, \bar{f}(H_j) + (1-\alpha)\, \bar{z}_{j-1} = \alpha\, \frac{1}{|H_j|} \sum_{h_t \in H_j} f(h_t) + (1-\alpha)\, \bar{z}_{j-1} \tag{5}$$

where $\bar{z}_j$ is the estimate of the mean context vector after the j-th update step, $\bar{f}(H_j)$ is the average context vector of the j-th mini-batch, and α is the smoothing factor for the EMA, with $0 < \alpha \leq 1$. In this work, we set $\alpha = 0.05$. The covariance matrix Σ_j is estimated similarly by computing the sample covariance matrix of the context vectors and updating it with EMA.

$$\Sigma_j = \alpha\, \frac{1}{|H_j| - 1} \sum_{h_t \in H_j} (f(h_t) - \bar{f}(H_j))(f(h_t) - \bar{f}(H_j))^\top + (1-\alpha)\, \Sigma_{j-1} \tag{6}$$

5. Experiments

5.1. Setup

To evaluate the effectiveness of the proposed Balanced Domain Randomization (BDR) method, we conduct experiments on two control tasks: Pendulum and Walker. We utilize the OpenAI Gymnasium [46] for the Pendulum task and the DeepMind Control (DMC) suite [47] for the Walker task. To test the adaptability and robustness of the Reinforcement Learning (RL) agents across varied dynamic conditions, we apply domain randomization facilitated by the CARL framework [19].

In the Pendulum task, the goal is to swing a pendulum from a hanging position to an upright position. The agent controls the torque applied to the pendulum, based on its position and angular velocity. The agent is penalized for deviations from the upright position as well as control costs. We introduce variability in the environment by altering the mass and length of the pendulum. For both training and evaluation, we use combinations of mass and length values from the set {0.1, 0.5, 1.0, 1.5, 2.0}, creating a broader range of scenarios to assess the agent's adaptability.

In the Walker task, the goal of a bipedal walker is to move forward efficiently while keeping balance. The agent controls the torques at its joints, based on its height, orientation, and velocity. The agent is rewarded for reaching the target velocity and maintaining balance. We alter the friction coefficients and the gravitational forces, using combinations of gravity values {1, 5, 10, 15, 20} and friction coefficients {0.1, 0.5, 1.0, 1.5, 2.0}.

In this framework, the context encoder maps the history of the agent's state–action pairs into a context vector using a Multi-Layer Perceptron (MLP) network. Here, we use five recent state–action pairs as we empirically find it sufficient to infer the environmental dynamics. This MLP consists of four layers, with 256, 128, 64, and 10 neurons, respectively. The context encoder outputs a context vector z_t, which represents a compressed embedding of the environmental dynamics. This vector is then fed into the policy network to condition the policy on the current environmental context. The context encoder is trained jointly with the contextual policy in an end-to-end manner to maximize the reward. For RL optimization, we employ the Twin Delayed Deep Deterministic policy gradient (TD3) algorithm [48].

5.2. Evaluation Metric

We evaluate the performance of the agents based on the episodic return, defined as the total sum of rewards accumulated over an episode. For a comprehensive evaluation, we run N episodes and aggregate the episodic returns by calculating the average for each domain. In this work, we run 100 episodes for each domain. Our evaluation focuses on both worst-case and average-case performance, using the following metrics:

$$\text{Minimum Episodic Return} = \min_{d \in D}(R_d)$$

The minimum episodic return metric captures the agent's performance in the most challenging domain, reflecting its robustness in difficult scenarios.

$$\text{Mean Episodic Return (Worst-}k\text{)} = \frac{1}{k}\sum_{i=1}^{k}(R_{\text{sorted},i})$$

The mean episodic return of the worst-k samples is calculated by first sorting all episodic returns in ascending order, from worst to best. The worst-performing k episodes are then selected, and their mean is computed. This metric provides insight into its resilience under unfavorable conditions, as it focuses on the episodes where the agent performs the worst.

$$\text{Mean Episodic Return} = \frac{1}{|D|}\sum_{d \in D}(R_d)$$

The mean episodic return indicates the average-case performance, highlighting how well the agent adapts to the variety of environments on average.

5.3. Results

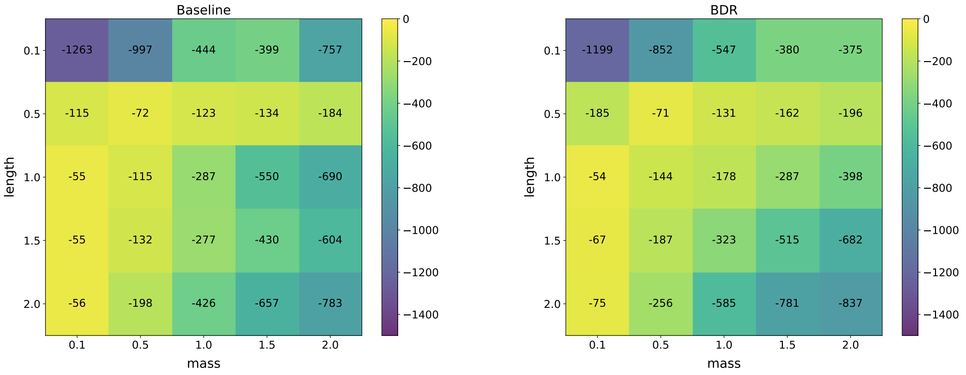
We evaluate the performance of the Uniform Domain Randomization (UDR) baseline and Balanced Domain Randomization (BDR) by keeping their configurations identical, including model architecture and hyperparameters, with the only distinction being the weighted RL loss component. Additionally, we compare the performance with Automatic Domain Randomization (ADR) [49], which gradually expands the randomization ranges during training. ADR realizes a training curriculum that gradually expands a distribution over training environments, from the initial range concentrated on a single environment.

Table 1 summarizes the quantitative results. In the Pendulum task, BDR improves the minimum episodic return by 5.11% (from −1263.97 to −1199.37), the mean episodic return (worst 10%) by 7.31% (from −1221.57 to −1132.30), and the mean episodic return by 3.45% (from −392.70 to −379.14), over the baseline. In the Walker task, BDR enhances the minimum episodic return by 24.81% (from 539.62 to 673.49), the mean episodic return (worst 10%) by 23.57% (from 516.80 to 638.59), and the mean episodic return by 0.79% (from 878.21 to 885.18). These consistent improvements across both tasks suggest that BDR reduces overfitting to common domains and enhances adaptability in rare scenarios.

Figures 3 and 4 show the average episodic return for each domain in the Pendulum and Walker tasks, respectively. We observe that the improvements are especially evident in domains with extreme conditions, such as the low length in the Pendulum task and the low gravity in the Walker task. Specifically, when the length is 0.1 in the Pendulum task, the BDR achieves an average episodic return of −670.6, outperforming the baseline, which exhibits an average return of −772. This represents a 13.13% improvement. When the gravity is 1 in the Walker task, the BDR improves an average episodic return by 11.38%, achieving an average episodic return of 763.4 while the baseline only achieves an average return of 685.4.

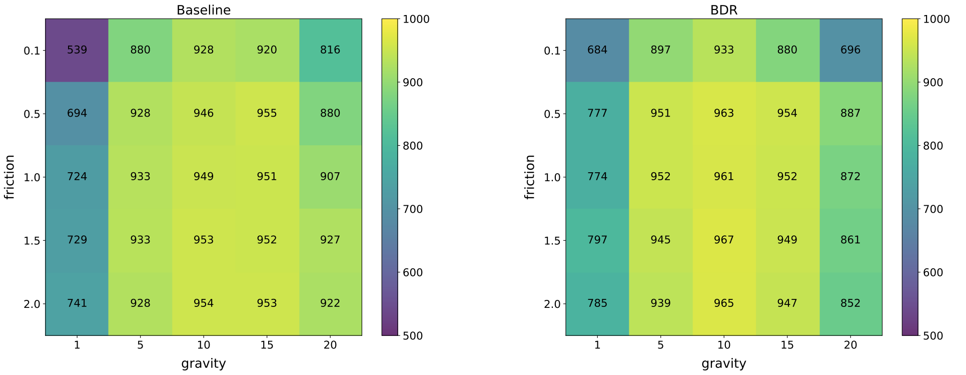
Table 1. Mean and minimum episodic returns in Pendulum and Walker tasks. The best performance is marked in bold.

Task	Method	Minimum Episodic Return	Mean Episodic Return (Worst 10%)	Mean Episodic Return
Pendulum	Baseline	−1263.97	−1221.57	−392.70 ± 319.62
	ADR	−1312.10	−1332.14	−569.46 ± 428.23
	BDR	**−1199.37**	**−1132.30**	**−379.14** ± 295.16
Walker	Baseline	539.62	516.80	878.21 ± 106.04
	ADR	224.65	220.40	560.88 ± 198.98
	BDR	**673.49**	**638.59**	**885.18** ± 88.89



(**a**) Baseline (**b**) BDR (Proposed)

Figure 3. Episodic returns of the baseline and BDR in the Pendulum task. The BDR (**b**) improves performance in domains with possibly rare dynamics, particularly when the pendulum length is 0.1, where the baseline policy (**a**) struggles.



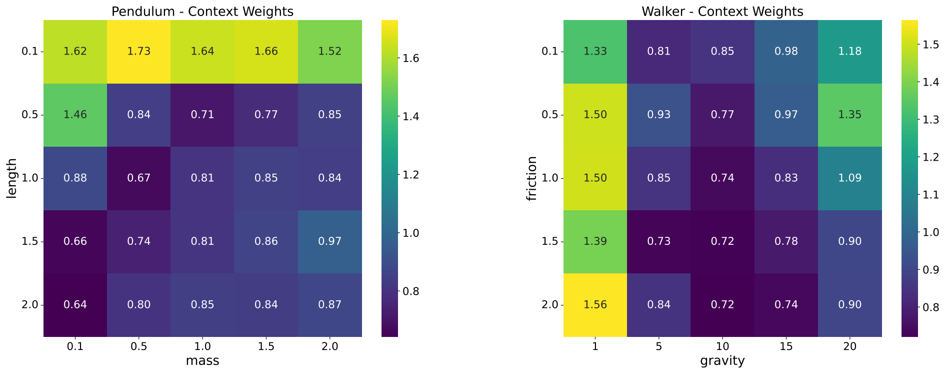
(**a**) Baseline (**b**) BDR (Proposed)

Figure 4. Episodic returns of the baseline and BDR in the Walker task. The BDR (**b**) improves performance, particularly when the gravity is 1, where the baseline policy (**a**) struggles.

We also find that ADR is not effective in learning a balanced behavior over diverse environmental contexts, as it does not improve the performance over the baseline both in the Pendulum and Walker tasks. Although there is still room for further improvement in worst-case scenarios, BDR effectively improves performance in challenging domains while maintaining its average-case performance.

5.4. Analysis

Weight assignment. We also analyze how BDR adjusts the weights for training samples from different domains. Figure 5 visualizes the average sample weights assigned to each domain. The analysis indicates that BDR gives higher weights to domains with smaller lengths in the Pendulum task (Figure 5a) and lower gravity in the Walker task (Figure 5b). It demonstrates that BDR effectively targets and emphasizes rare domains where the baseline performs poorly by measuring the rarity in the embedding space.



(**a**) Pendulum (**b**) Walker

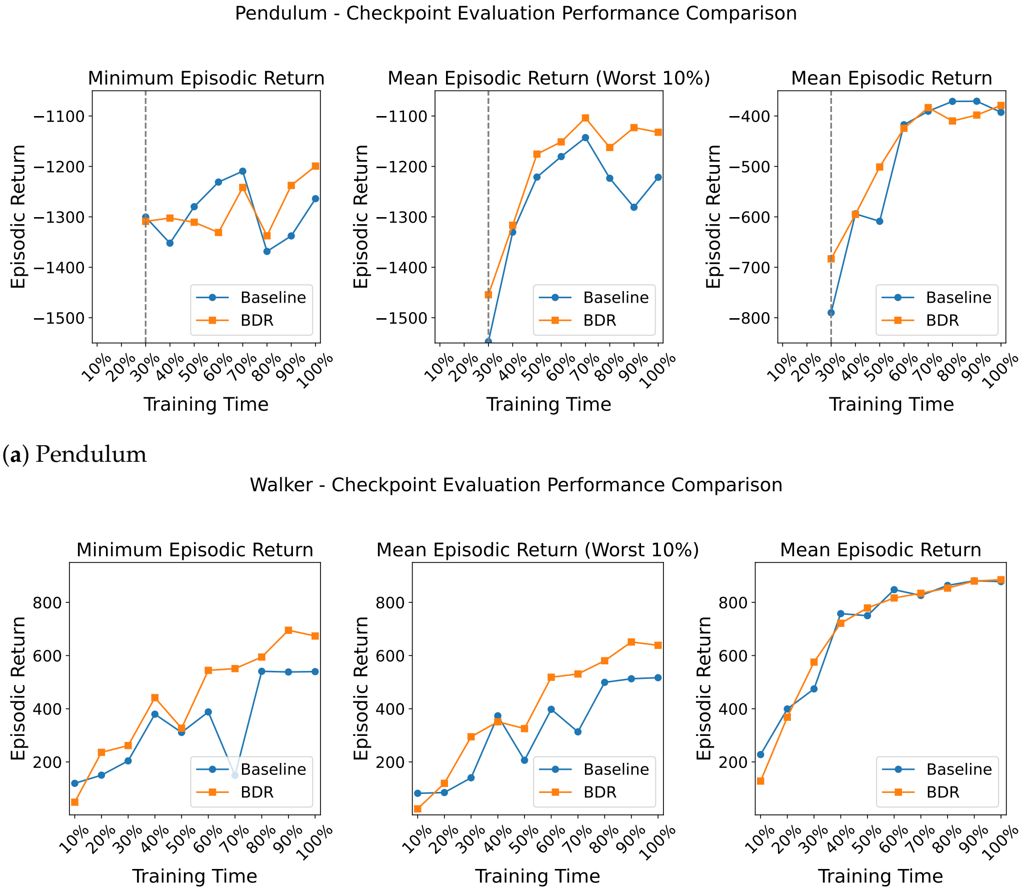
Figure 5. Average sample weights for each domain. The BDR framework assigns higher weights to rare domains, when the length is 0.1 in the Pendulum task (**a**) and when the gravity is 1 in the Walker task (**b**), ensuring these underrepresented domains receive more attention during training.

Finally, we assess the efficiency of BDR by observing performance changes throughout the training process. Figure 6 illustrates the differences in learning performance between the two methods at each training checkpoint. The results demonstrate that BDR efficiently optimizes worst-case performance while maintaining average-case performance, highlighting its robustness and reliability in varied conditions during training.

Balancing parameter. To further investigate the impact of the balancing parameter β, we conducted an ablation study by varying its value from 0 (which denotes uniform randomization) to 1 (which places full emphasis on rare domains). Table 2 shows that $\beta = 0.5$ yields the best performance in most cases. However, performance degrades for higher values of β, since overemphasizing rare domains may hinder the agent learning its general strategies in more common scenarios. Although it may be necessary to appropriately tune the value of β, depending on tasks, the empirical results show that $\beta = 0.5$ provides the best trade-off between learning from rare and common scenarios in both tasks.

Table 2. Mean and minimum episodic returns of the BDR with different values of β. The best performance is marked in bold.

Task	Method	Minimum Episodic Return	Mean Episodic Return (Worst 10%)	Mean Episodic Return
Pendulum	Baseline ($\beta = 0$)	-1263.97	-1221.57	-392.70 ± 319.62
	$\beta = 0.25$	−1299.66	−1244.98	−450.21 ± 317.04
	$\beta = 0.5$	−**1199.37**	−**1132.30**	−**379.14** ± 295.16
	$\beta = 0.75$	−1227.09	−1134.78	−411.02 ± 319.18
	$\beta = 1$	−1386.99	−1186.47	−464.74 ± 323.14
Walker	Baseline ($\beta = 0$)	539.62	516.80	878.21 ± 106.04
	$\beta = 0.25$	668.35	633.97	**900.73** ± 85.59
	$\beta = 0.5$	**673.49**	**638.59**	885.18 ± 88.89
	$\beta = 0.75$	454.86	455.92	767.93 ± 109.67
	$\beta = 1$	497.80	567.64	851.82 ± 106.13



(**a**) Pendulum

(**b**) Walker

Figure 6. Performance trends of the baseline and BDR over training time. In both the Pendulum and Walker tasks, BDR consistently outperforms the baseline in worst-case performance (minimum episodic return and mean episodic return in worst 10%) while maintaining average-case performance (mean episodic return) during training.

5.5. Evaluation in Safety-Critical Navigation Tasks

To evaluate the efficacy of the BDR framework in more complex and safety-critical scenarios, we adopt a 2D safe navigation task using the SafetyGymnasium environment [50]. This environment introduces hazards as well as goal-reaching objectives, representing safety-critical real-world challenges such as autonomous navigation and robotic deployment. The objective is to reach a randomly generated goal while avoiding randomly placed unsafe regions (hazards). The agent controls actuators for linear and rotational velocity, receiving rewards for reaching the goal and penalties for entering unsafe regions. To introduce variability in dynamics, we alter both the mass and the moment of inertia (MOI). For training and evaluation, we use combinations of mass and MOI from the set {0.1, 0.3, 0.5, 0.7, 1.0}. As well as episodic return, we measure episodic cost as a safety-specific metric. The episodic cost is measured as accumulated penalties for entering unsafe regions.

Table 3 reports the quantitative results. BDR improves the minimum episodic return by 500% (from 0.12 to 0.72), the mean episodic return (worst 10%) by 80.77% (from -1.82 to -0.35), and the mean episodic return by 251.60% (from 2.81 to 9.88), over the baseline. For safety-specific evaluation, BDR reduces the minimum episodic cost by 25.35% (from 109.97 to 82.09), the mean episodic cost (worst 10%) by 34.46% (from 484.53 to 317.56), and the mean episodic cost by 9.62% (from 61.62 to 55.69), over the baseline. These results demonstrate the efficacy of BDR in reducing safety risks while maintaining task performance.

Table 3. Mean and minimum episodic returns and costs in the safe navigation task. The best performance is marked in bold.

Task	Method	Minimum Episodic Return	Mean Episodic Return (Worst 10%)	Mean Episodic Return
Navigation	Baseline	0.12	−1.82	2.81 ± 1.37
	BDR	**0.72**	−**0.35**	**9.88** ± 4.84
		Maximum Episodic Cost	**Mean Episodic Cost (Worst 10%)**	**Mean Episodic Cost**
	Baseline	109.97	484.53	61.62 ± 16.96
	BDR	**82.09**	**317.56**	**55.69** ± 11.09

6. Conclusions

This paper presents Balanced Domain Randomization (BDR) to address the imbalance problem in contextual Reinforcement Learning (RL) by prioritizing the agent to learn its policy in rare domains during training. Our key idea is to automatically identify insufficiently learned domains by evaluating context rarity based on the statistics of context vectors. BDR leverages Mahalanobis distance in the context embedding space to assess rarity and adaptively adjust the training weights, ensuring sufficient exposure to rare scenarios. Our experimental results on the Pendulum and Walker tasks demonstrate that BDR improves worst-case performance, with a 5.11% gain over the baseline in Pendulum and 24.81% in Walker, while maintaining average performance. BDR has the potential to enhance the robustness of RL agents for real-world applications, such as robotics and autonomous driving, by improving sim-to-real transfer and reducing the risk of failure in unpredictable scenarios. While some hyperparameter tuning is required, this could be further explored in future work. Moreover, we expect that BDR's generalization capabilities can be extended to entirely unseen domains.

Author Contributions: Conceptualization, C.K., W.C. and J.C.; methodology, C.K. and W.C.; software, C.K.; validation, C.K. and W.C.; formal analysis, C.K. and W.C.; investigation, C.K. and W.C.; resources, C.K., W.C. and J.C.; data curation, C.K.; writing—original draft preparation, C.K. and W.C.; writing—review and editing, C.K., W.C. and J.C.; visualization, C.K.; supervision, J.C.; project administration, J.C.; funding acquisition, J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Unmanned Swarm CPS Research Laboratory Program of Defense Acquisition Program Administration and Agency for Defense Development (UD220005VD).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data and code are available at https://github.com/CheongWoong/CARL_analysis (accessed on 15 August 2024).

Conflicts of Interest: Author Jaesik Choi was employed by the company INEEJI Corporation. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RL	Reinforcement Learning
UDR	Uniform Domain Randomization
ADR	Automatic Domain Randomization
BDR	Balanced Domain Randomization
MDP	Markov Decision Process
POMDPs	Partially Observable MDPs

EMA	Exponential Moving Average
DMC	DeepMind Control
MLP	Multi-Layer Perceptron
TD3	Twin Delayed Deep Deterministic policy gradient

References

1. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef] [PubMed]
2. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:1509.02971.
3. Levine, S.; Pastor, P.; Krizhevsky, A.; Ibarz, J.; Quillen, D. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *Int. J. Robot. Res.* **2018**, *37*, 421–436. [CrossRef]
4. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of go without human knowledge. *Nature* **2017**, *550*, 354–359. [CrossRef]
5. Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature* **2020**, *588*, 604–609. [CrossRef]
6. Vinyals, O.; Babuschkin, I.; Czarnecki, W.M.; Mathieu, M.; Dudzik, A.; Chung, J.; Choi, D.H.; Powell, R.; Ewalds, T.; Georgiev, P.; et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* **2019**, *575*, 350–354. [CrossRef]
7. Ellis, B.; Cook, J.; Moalla, S.; Samvelyan, M.; Sun, M.; Mahajan, A.; Foerster, J.; Whiteson, S. Smacv2: An improved benchmark for cooperative multi-agent reinforcement learning. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 10–15 December 2024.
8. Gu, S.; Holly, E.; Lillicrap, T.; Levine, S. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017.
9. Andrychowicz, O.M.; Baker, B.; Chociej, M.; Jozefowicz, R.; McGrew, B.; Pachocki, J.; Petron, A.; Plappert, M.; Powell, G.; Ray, A.; et al. Learning dexterous in-hand manipulation. *Int. J. Robot. Res.* **2020**, *39*, 3–20. [CrossRef]
10. Patel, U.; Kumar, N.K.S.; Sathyamoorthy, A.J.; Manocha, D. Dwa-rl: Dynamically feasible deep reinforcement learning policy for robot navigation among mobile obstacles. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021.
11. Zhu, K.; Zhang, T. Deep reinforcement learning based mobile robot navigation: A review. *Tsinghua Sci. Technol.* **2021**, *26*, 674–691. [CrossRef]
12. Tobin, J.; Fong, R.; Ray, A.; Schneider, J.; Zaremba, W.; Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017.
13. Yu, W.; Tan, J.; Liu, C.K.; Turk, G. Preparing for the unknown: Learning a universal policy with online system identification. In Proceedings of the Robotics: Science and Systems (RSS), Cambridge, MA, USA, 12–16 July 2017.
14. Peng, X.B.; Andrychowicz, M.; Zaremba, W.; Abbeel, P. Sim-to-real transfer of robotic control with dynamics randomization. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018.
15. Tan, J.; Zhang, T.; Coumans, E.; Iscen, A.; Bai, Y.; Hafner, D.; Bohez, S.; Vanhoucke, V. Sim-to-Real: Learning Agile Locomotion For Quadruped Robots. In Proceedings of the Robotics: Science and Systems (RSS), Pittsburgh, PA, USA, 26–30 June 2018.
16. Chen, X.; Hu, J.; Jin, C.; Li, L.; Wang, L. Understanding Domain Randomization for Sim-to-real Transfer. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 4 May 2021.
17. Luo, F.M.; Jiang, S.; Yu, Y.; Zhang, Z.; Zhang, Y.F. Adapt to Environment Sudden Changes by Learning a Context Sensitive Policy. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), Virtual, 22 February–1 March 2022.
18. Hallak, A.; Di Castro, D.; Mannor, S. Contextual markov decision processes. *arXiv* **2015**, arXiv:1502.02259.
19. Benjamins, C.; Eimer, T.; Schubert, F.; Mohan, A.; Döhler, S.; Biedenkapp, A.; Rosenhahn, B.; Hutter, F.; Lindauer, M. Contextualize Me—The Case for Context in Reinforcement Learning. *arXiv* **2022**, arXiv:2202.04500.
20. Rakelly, K.; Zhou, A.; Finn, C.; Levine, S.; Quillen, D. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In Proceedings of the International Conference on Machine Learning (ICML), Long Beach, CA, USA, 10–15 June 2019.
21. Zhou, W.; Pinto, L.; Gupta, A. Environment probing interaction policies. In Proceedings of the International Conference on Learning Representations (ICLR), New Orleans, LA, USA, 6–9 May 2019.
22. Lee, K.; Seo, Y.; Lee, S.; Lee, H.; Shin, J. Context-aware dynamics model for generalization in model-based reinforcement learning. In Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 12–18 July 2020.
23. Seo, Y.; Lee, K.; Clavera Gilaberte, I.; Kurutach, T.; Shin, J.; Abbeel, P. Trajectory-wise multiple choice learning for dynamics generalization in reinforcement learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 12968–12979.
24. Packer, C.; Gao, K.; Kos, J.; Krähenbühl, P.; Koltun, V.; Song, D. Assessing generalization in deep reinforcement learning. *arXiv* **2018**, arXiv:1810.12282.
25. Zhao, W.; Queralta, J.P.; Westerlund, T. Sim-to-real transfer in deep reinforcement learning for robotics: A survey. In Proceedings of the Symposium Series on Computational Intelligence, Canberra, Australia, 1–4 December 2020.

26. Kirk, R.; Zhang, A.; Grefenstette, E.; Rocktäschel, T. A survey of zero-shot generalisation in deep reinforcement learning. *J. Artif. Intell. Res.* **2023**, *76*, 201–264. [CrossRef]
27. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [CrossRef]
28. Drummond, C.; Holte, R.C. C4. 5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling. In Proceedings of the Workshop on Learning from Imbalanced Datasets II, Washington, DC, USA, 21 August 2003.
29. Estabrooks, A.; Jo, T.; Japkowicz, N. A multiple resampling method for learning from imbalanced data sets. *Comput. Intell.* **2004**, *20*, 18–36. [CrossRef]
30. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
31. Ren, M.; Zeng, W.; Yang, B.; Urtasun, R. Learning to reweight examples for robust deep learning. In Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 25–31 July 2018.
32. Cao, K.; Wei, C.; Gaidon, A.; Arechiga, N.; Ma, T. Learning imbalanced datasets with label-distribution-aware margin loss. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), Vancouver, BC, Canada, 8–14 December 2019.
33. Cui, Y.; Jia, M.; Lin, T.Y.; Song, Y.; Belongie, S. Class-balanced loss based on effective number of samples. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
34. Alshammari, S.; Wang, Y.X.; Ramanan, D.; Kong, S. Long-tailed recognition via weight balancing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022.
35. Liu, S.; Garrepalli, R.; Dietterich, T.; Fern, A.; Hendrycks, D. Open category detection with PAC guarantees. In Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 25–31 July 2018.
36. Yin, X.; Yu, X.; Sohn, K.; Liu, X.; Chandraker, M. Feature transfer learning for face recognition with under-represented data. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
37. Bengio, Y.; Louradour, J.; Collobert, R.; Weston, J. Curriculum learning. In Proceedings of the International Conference on Machine Learning (ICML), Montreal, QC, Canada, 14–18 June 2009.
38. Narvekar, S.; Peng, B.; Leonetti, M.; Sinapov, J.; Taylor, M.E.; Stone, P. Curriculum learning for reinforcement learning domains: A framework and survey. *J. Mach. Learn. Res.* **2020**, *21*, 1–50.
39. Kim, S.; Lee, K.; Choi, J. Variational Curriculum Reinforcement Learning for Unsupervised Discovery of Skills. In Proceedings of the International Conference on Machine Learning (ICML), Honolulu, HI, USA, 23–29 July 2023.
40. Naeem, M.F.; Oh, S.J.; Uh, Y.; Choi, Y.; Yoo, J. Reliable fidelity and diversity metrics for generative models. In Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 12–18 July 2020.
41. Han, J.; Choi, H.; Choi, Y.; Kim, J.; Ha, J.W.; Choi, J. Rarity Score: A New Metric to Evaluate the Uncommonness of Synthesized Images. In Proceedings of the International Conference on Learning Representations (ICLR), Kigali, Rwanda, 1–5 May 2023.
42. Smallwood, R.D.; Sondik, E.J. The optimal control of partially observable Markov processes over a finite horizon. *Oper. Res.* **1973**, *21*, 1071–1088. [CrossRef]
43. Kaelbling, L.P.; Littman, M.L.; Cassandra, A.R. Planning and acting in partially observable stochastic domains. *Artif. Intell.* **1998**, *101*, 99–134. [CrossRef]
44. Mahalanobis, P.C. On the generalized distance in statistics. *Sankhyā Indian J. Stat. Ser. A* **2018**, *80*, S1–S7.
45. Dragicevic, A.Z. Spacetime discounted value of network connectivity. *Adv. Complex Syst.* **2018**, *21*, 1850018. [CrossRef]
46. Towers, M.; Kwiatkowski, A.; Terry, J.K.; Balis, J.U.; De Cola, G.; Deleu, T.; Goulão, M.; Kallinteris, A.; KG, A.; Krimmel, M.; et al. Gymnasium: A standard interface for reinforcement learning environments. *arXiv* **2024**, arXiv:2407.17032.
47. Tunyasuvunakool, S.; Muldal, A.; Doron, Y.; Liu, S.; Bohez, S.; Merel, J.; Erez, T.; Lillicrap, T.; Heess, N.; Tassa, Y. dm_control: Software and tasks for continuous control. *Softw. Impacts* **2020**, *6*, 100022. [CrossRef]
48. Fujimoto, S.; Hoof, H.; Meger, D. Addressing function approximation error in actor-critic methods. In Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 25–31 July 2018.
49. Akkaya, I.; Andrychowicz, M.; Chociej, M.; Litwin, M.; McGrew, B.; Petron, A.; Paino, A.; Plappert, M.; Powell, G.; Ribas, R.; et al. Solving rubik's cube with a robot hand. *arXiv* **2019**, arXiv:1910.07113.
50. Ji, J.; Zhang, B.; Zhou, J.; Pan, X.; Huang, W.; Sun, R.; Geng, Y.; Zhong, Y.; Dai, J.; Yang, Y. Safety gymnasium: A unified safe reinforcement learning benchmark. In Proceedings of the Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, 10–16 December 2023.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.