Sliding Puzzles Gym: A Scalable Benchmark for State Representation in Visual Reinforcement Learning

Bryan L. M. de Oliveira^{*1,2} **Murilo L. da Luz**^{1,2} **Bruno Brandão**^{1,2}

Luana G. B. Martins^{1,2} **Telma W. de L. Soares**^{1,2} **Luckeciano C. Melo**^{1,3}

^{1}AKCIT ^{2}Federal University of Goiás ^{3}OATML, University of Oxford

Abstract

Learning effective visual representations is crucial in open-world environments where agents encounter diverse and unstructured observations. This ability enables agents to extract meaningful information from raw sensory inputs, like pixels, which is essential for generalization across different tasks. However, evaluating representation learning separately from policy learning remains a challenge in most reinforcement learning (RL) benchmarks. To address this, we introduce the Sliding Puzzles Gym (SPGym), a benchmark that extends the classic 15-tile puzzle with variable grid sizes and observation spaces, including large real-world image datasets. SPGym allows scaling the representation learning challenge while keeping the latent environment dynamics and algorithmic problem fixed, providing a targeted assessment of agents' ability to form compositional and generalizable state representations. Our experiments with both model-free and model-based RL algorithms, with and without explicit representation learning components, show that as the representation challenge scales, SPGym effectively distinguishes agents based on their capabilities. Moreover, SPGym reaches difficulty levels where no tested algorithm consistently excels, highlighting key challenges and opportunities for advancing representation learning for decision-making research.

1 Introduction

Learning meaningful representations from raw sensory inputs, such as visual data, is fundamental to reinforcement learning (RL) agents' ability to generalize across different tasks in complex, open-world environments. In visual RL, agents must process high-dimensional pixel data, extract useful features, and utilize these features for decision-making. This becomes especially crucial as real-world applications demand adaptability to unstructured and diverse observations. However, measuring an agent's representation learning capabilities independently from other learning tasks, such as policy optimization or dynamics modeling, remains a key challenge in RL benchmarks.

Widely adopted RL benchmarks like Atari [2], DeepMind Control Suite [16], and DeepMind Lab [1] focus on evaluating overall agent performance, where representation learning occurs alongside policy optimization and dynamics modeling. While specialized benchmarks have emerged to address specific aspects of visual learning — such as the Distracting Control Suite [15] for robustness to visual noise and ProcGen [3] for generalization through procedural generation — their primary goals do not include isolating representation learning capabilities. This makes it challenging to evaluate an agent's ability to form robust state representations independently from other learning aspects.

[*] Correspondence to: bryanlincoln@discente.ufg.br.

1st Workshop on Open-World Agents at the 38th Conference on Neural Information Processing Systems, 2024.

We address this gap by introducing the **Sliding Puzzles Gym (SPGym)**,[2] a benchmark designed to isolate and evaluate visual representation learning capabilities in RL agents. Unlike existing benchmarks where representation learning intertwines with policy optimization and dynamics modeling, SPGym provides a controlled environment where the underlying task structure and dynamics remain fixed while the visual complexity can scale systematically. By extending the classic 15-tile puzzle with variable grid sizes and rich observation spaces, such as image-based tiles, SPGym enables researchers to precisely measure how well agents can form robust state representations independent of their policy learning abilities.

Our experiments evaluate the performance of both model-free and model-based RL agents on SPGym, assessing their ability to handle increasingly complex visual input spaces. We compare standard PPO [14] and DreamerV3 [4] agents with variants that modify their representation learning components. The results demonstrate that, as the image pool size increases, SPGym effectively differentiates between agents based on their representation learning capabilities. Specifically, model-based agents like DreamerV3 [4] outperform model-free ones like PPO [14] in most scenarios, though even these methods struggle with higher levels of visual complexity.

Contributions. We summarize our main contributions as follows:

- We introduce the Sliding Puzzles Gym (SPGym), a scalable benchmark for assessing representation learning in visual decision-making by allowing systematic scaling of visual complexity while maintaining fixed environment dynamics;
- Through experiments with model-free and model-based agents, we show that SPGym effectively differentiates algorithms based on their representation learning capabilities; and
- We reveal limitations in current RL methods' ability to handle increasingly complex and diverse visual inputs, showcasing the scalability and challenge SPGym offers as a benchmark.

Through SPGym, we provide a focused benchmark that isolates representation learning challenges from policy learning, enabling targeted research into visual RL methods. This separation allows researchers to systematically improve agents' ability to handle complex visual inputs, a critical capability for real-world applications.

2 Related Work

Reinforcement learning benchmarks are essential tools for evaluating agent performance across various tasks. Popular benchmarks like Atari [2], DeepMind Control Suite [16], and DeepMind Lab [1] primarily assess overall agent performance, inherently combining representation learning with policy optimization and dynamics modeling. This integration makes it challenging to isolate the specific impact of representation learning on an agent's performance. Even methods designed to enhance representation learning for RL agents, such as DARLA [5], CURL [8], RAD [9], DrQ [7], CBM [10], and CycAug [11], are typically evaluated within these entangled settings, preventing a clear, isolated assessment of their representation learning capabilities.

Among existing benchmarks, ProcGen [3] is one of the closest to our work. ProcGen takes a step towards evaluating visual generalization through diversity by offering procedurally generated levels that challenge agents to adapt to unseen environments. However, it does not provide a controlled way to isolate representation learning from policy and dynamics learning. In ProcGen, the complexity of procedurally generated levels can obscure the specific contributions of representation learning, as agents must simultaneously learn representations, policies, and environment dynamics. This entanglement makes it challenging to pinpoint whether performance improvements stem from better representations or other factors such as improved policy learning or exploration strategies.

Alternatively, the Distracting Control Suite [15], an extension of DM Control, introduces visual distractions to evaluate agents' robustness to irrelevant variations. While it allows for parametrizable control over distraction complexity, its primary focus is on agents' ability to *ignore* these distractions rather than to *extract* meaningful information from visually complex observations, assessing resilience to noise but not the capability to learn and utilize rich visual representations.

[2] Code available at `https://github.com/bryanoliveira/sliding-puzzles-gym`.

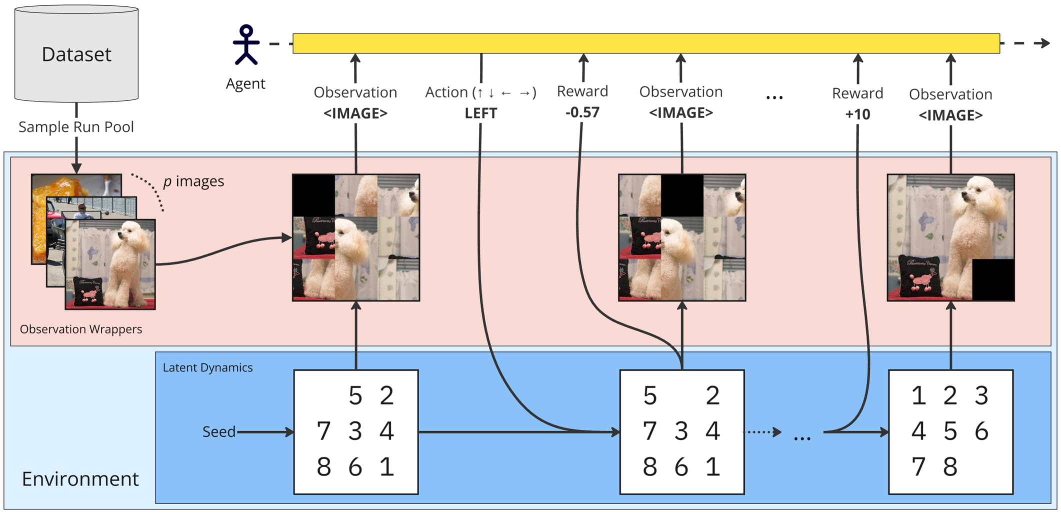

Figure 1: **Overview of the Sliding Puzzles Gym (SPGym).** The framework extends the 15-tile puzzle by incorporating image-based tiles, allowing scalable representation complexity while maintaining fixed environment dynamics. Agents must solve the puzzle using only image observations.

SPGym addresses these limitations by providing a controlled environment where difficulty arises solely from the complexity of visual observations, rather than changes in task dynamics or objectives. This design allows for a more precise evaluation of an agent's representation learning capabilities. By systematically scaling the visual complexity while keeping the underlying task structure and dynamics fixed, SPGym enables researchers to isolate and measure the impact of representation learning independently from other learning aspects. This makes SPGym particularly well-suited for comparing RL algorithms that specifically target improvements in representation learning.

3 The Sliding Puzzles Gym

The Sliding Puzzles Gym (SPGym) extends the 15-tile puzzle, where players must rearrange a shuffled 4×4 grid of numbered tiles into the correct sequence using four actions: *UP*, *DOWN*, *LEFT*, or *RIGHT*. SPGym generalizes this concept by supporting variable grid sizes, from 2×2 to larger $H \times W$ configurations, and incorporating diverse observation spaces. Here, we focus on 3×3 grids to emphasize visual representation learning, providing only image-based observations to the agent.

Environment Dynamics and Actions. At each step, the agent can slide a tile adjacent to the empty space into that position, making the action space discrete and deterministic, as illustrated in Figure 1. The objective is to rearrange shuffled image tiles into their correct positions, forming a complete, intelligible image. While the underlying puzzle mechanics remain simple and consistent, the visual representation challenge scales with the diversity of images used. Agents must learn to extract meaningful features from potentially complex visual inputs, recognize patterns across different images, and understand how these features relate to the puzzle's solution state. This separation between fixed dynamics and scalable visual complexity allows SPGym to isolate and evaluate an agent's representation learning capabilities independently from its policy learning abilities.

Observation Spaces and Wrappers. SPGym's internal representation is a 2D array of tile indices representing the current grid state. We provide flexible wrappers that convert states into various observation modalities, ranging from simple one-hot encodings to complex visual observations, such as real-world images and text (Figure 2). This adaptability makes SPGym a versatile benchmark for exploring how reinforcement learning agents handle diverse observation spaces. In this paper, we focus on visual representation learning, leaving other modalities for future exploration.

For visual observations, we employ image overlays. In each run, we sample p images from a predefined dataset to form a pool. SPGym is dataset-agnostic, allowing the use of any image dataset, including procedurally generated images. At the start of each episode, we select a random image

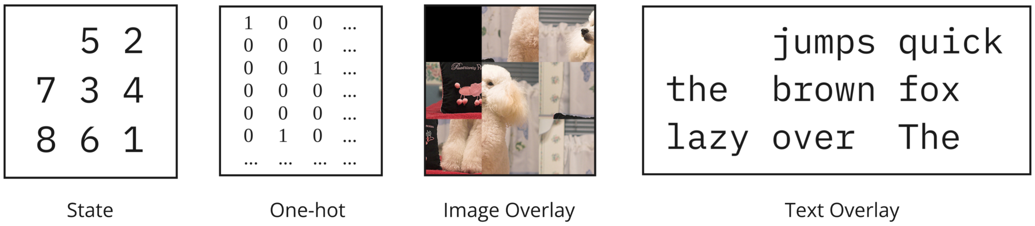

Figure 2: **Different observation modalities in SPGym.** Each modality presents a unique challenge for representation learning. The four presented observations represent the same latent puzzle state.

from the pool, split it into $H \times W$ indexed patches, and overlay it onto the puzzle. The agent's task is to reconstruct the shuffled image, testing its ability to form compositional representations from pixels.

To initialize the puzzle's state, SPGym offers two methods. The primary method, used in this paper, generates a random $H \times W$ array and ensures solvability by adjusting the parity of the puzzle, if necessary, by swapping the first two tiles [6]. The second method, which can facilitate curriculum learning in future work, begins with a solved puzzle and applies a series of random valid moves to create an initial state, allowing for the selection of easier starting configurations.

Reward Function. At each time step, the reward is computed as follows:

$$r_t = \begin{cases} -\frac{\sum_{i,j} |x_{i,j} - x^*_{i,j}| + |y_{i,j} - y^*_{i,j}|}{D}, & \text{if action is valid} \\ -1, & \text{if action is invalid} \\ +10, & \text{if puzzle is solved} \end{cases} \text{, with} \tag{1}$$

$$D = \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} [(i,j) \neq (H-1, W-1)] \cdot (\max(i, H-1-i) + \max(j, W-1-j)) \, . \tag{2}$$

Here, $(x_{i,j}, y_{i,j})$ represents the current position of the tile at index (i, j), $(x^*_{i,j}, y^*_{i,j})$ is its target position, and D is the maximum possible sum of Manhattan distances, excluding the blank tile. Invalid actions, such as attempting to move a tile outside the grid boundaries, result in a penalty of -1. For example, in the puzzle shown in Figure 2, the *DOWN* and *RIGHT* actions would be invalid. Successfully solving the puzzle rewards the agent with $+10$, and the episode terminates.

Complexity Scalability. A key feature of our benchmark is its scalable difficulty. SPGym increases representation learning complexity by keeping the grid size fixed while expanding the pool of images. This approach holds the underlying puzzle dynamics constant, ensuring that the increased difficulty comes solely from the agent's need to handle a larger variety of visual observations.

SPGym also scales the challenge by adjusting grid sizes. Larger grids increase the search space for solving the puzzle, though the core algorithmic problem remains unchanged.[3] Larger grid sizes also increase the complexity of representation learning by splitting images into more patches. Although scaling the grid size helps evaluate the efficacy of policy-learning algorithms in more challenging settings, we find that a 3 × 3 grid sufficiently differentiates the performance of tested algorithms.

By maintaining fixed environment dynamics across different difficulty levels, SPGym ensures that performance variations reflect the agent's ability to learn and generalize visual representations, rather than adapt to changing dynamics. Consequently, SPGym rigorously assesses an agent's capability to handle increasingly complex visual inputs while preserving consistent underlying behaviors.

[3] The same algorithm can solve the puzzle regardless of grid size [12, 17].

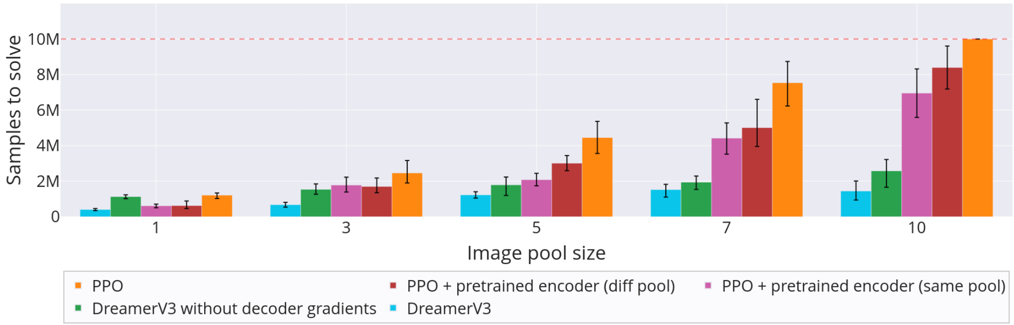

Figure 3: **Number of environment samples to solve the puzzle as a function of the pool size (lower is better).** We consider the puzzle solved when the agents reach 80% success rate and cap the number of environment steps per run to 10 million. Error bars represent the 95% confidence interval, calculated using bootstrap resampling with 1,000 iterations from 5 independent seeds.

4 Experimental Setup

We evaluate the performance of both model-free and model-based reinforcement learning agents using the image-based variation of SPGym. To simulate open-world conditions, we select images from the *validation* split of the ImageNet-1k [13] dataset.[4] This choice allows future work to leverage encoders pretrained on the *train* split, ensuring reproducibility and comparability with our results.

Agents and Variants. For model-free experiments, we employ Proximal Policy Optimization [14, PPO] agents. In the model-based category, we use agents based on the DreamerV3 [4] architecture, which includes an autoencoder to enhance representation learning. To evaluate how well SPGym differentiates agents based on their representation learning capabilities, we compare the performance of standard PPO and DreamerV3 agents with variants that alter the representation learning process. Specifically, we include a PPO variant with pretrained encoders and a DreamerV3 variant without gradients from the decoder (by setting the decoder's *loss_scale* to 0). For the pretrained PPO agents, we use encoders from agents trained on both the *same pool* and *diff*erent *pool*s of images. This allows us to assess potential improvements in sample efficiency with better encoders and the generalization capabilities of these pretrained models to new visual inputs. We load encoders pretrained on the same pool sizes and leave the exploration of how encoders trained with larger pool sizes generalize to unseen pools for future work. In all configurations, gradients propagate through the entire model.

Experimental Details. We measure the agents' performance primarily through sample efficiency, defined as the number of environment interactions required to reach a predefined threshold of 80% success rate. To prevent excessively long runs, we impose a cap of 10 million environment steps per run, and we apply early stopping if the agents maintain a success rate of 100% for at least 100 episodes. We also limit the agent's step count per episode to 1,000. Appendix A contains all hyperparameters relevant for result reproduction.

We conduct all experiments using 5 independent seeds and report the average results with corresponding error bars. For Figure 3, error bars represent the 95% confidence interval calculated using bootstrap resampling with 1,000 iterations. In Figure 4, we employ a rolling window average over 100,000 environment steps per run to account for potentially varying log rates across different runs. The 95% confidence interval for this figure is computed using the Z-score method.

Our hardware setup comprises an AMD Ryzen 7 3700X CPU, an NVIDIA RTX 3090 GPU, 64GB of RAM, and 128GB of swap space. Using this configuration, the most time-consuming DreamerV3 runs require approximately 20 hours, while the longest PPO runs complete in around 30 minutes.[5]

[4] Available at `https://huggingface.co/datasets/ILSVRC/imagenet-1k`.

[5] This runtime disparity is primarily due to DreamerV3's extensive use of swap space for its replay buffer.

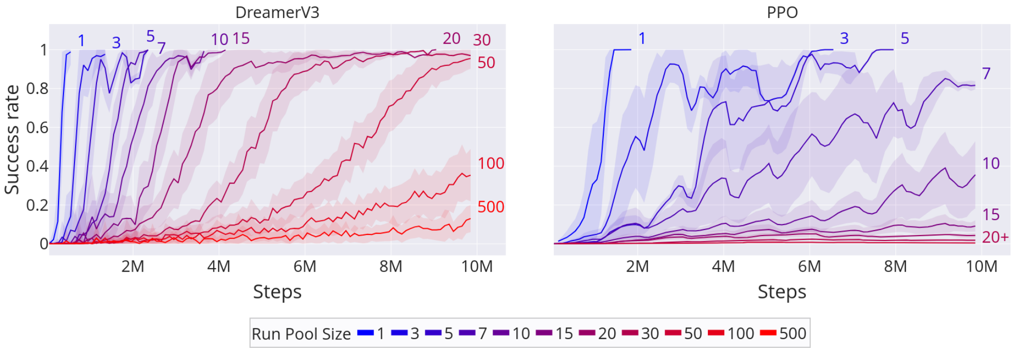

Figure 4: **Success rate as a function of environment steps.** The gradual increase in representation complexity affects the sample efficiency of standard PPO and DreamerV3 agents. Error bands indicate the 95% confidence interval, derived from 5 independent seeds and a rolling window of 100,000 environment steps, using the Z-score method.

5 Results

In this section, we address the following key research questions through our experiments:

Can SPGym distinguish agents based on their representation learning performance? Figure 3 demonstrates that SPGym effectively differentiates agents based on their representation learning capabilities. Model-based agents, especially the standard DreamerV3, consistently outperform model-free agents as the image pool complexity increases. This indicates that SPGym not only assesses the agents' task-solving abilities but also their skill in learning and applying robust state representations. The significant performance gap underscores the critical role of advanced representation learning mechanisms in handling complex environments.

How do representation-learning specific components impact agent performance in SPGym? Our analysis indicates that representation-learning components benefit agent performance. The DreamerV3 agent, with its integrated autoencoder structure, outperforms its variant without the decoder. This underscores the role of autoencoders in enhancing the agent's ability to abstract and generalize from visual inputs, which is crucial in environments with high visual complexity. The presence of these components allows agents to form more nuanced and effective representations, directly impacting their decision-making efficiency.

Does pretraining encoders improve the performance of model-free agents? Pretraining encoders significantly enhances the performance of PPO agents, enabling them to outperform their non-pretrained counterparts, even when the pretraining occurs on different image pools. This suggests that pretrained encoders capture essential, transferable features that benefit learning across various tasks. It highlights the potential of leveraging pretraining as a strategy to enhance the adaptability and efficiency of model-free agents in diverse environments.

How does the complexity of the image pool affect the performance of different agents? Figure 4 shows that increasing the image pool size leads to a decline in sample efficiency for both DreamerV3 and PPO agents. This trend highlights the challenges in scaling representation learning for more complex visual inputs. Even DreamerV3, which generally excels, struggles with larger pool sizes, indicating that current techniques may not fully capture the intricacies of diverse environments. This underscores SPGym's potential to push the boundaries of current methodologies and inspire more advanced representation learning strategies.

Our experiments confirm that SPGym effectively differentiates agents based on their representation learning capabilities. The benchmark's increasing difficulty, driven by visual diversity, highlights ongoing challenges in representation learning for decision-making tasks. These results suggest promising directions for future research, particularly in developing agents capable of handling more complex and diverse observation spaces.

6 Conclusion

We introduce the Sliding Puzzles Gym (SPGym), a scalable and flexible benchmark designed to evaluate the representation learning capabilities of reinforcement learning (RL) algorithms. By decoupling the complexity of the representation challenge from the underlying task structure, SPGym provides a controlled setting to assess agents' ability to form compositional and generalizable state representations. Our experiments highlight the effectiveness of approaches like DreamerV3, which outperforms PPO agents in most settings. However, even these advanced methods struggle as the complexity of the puzzle increases, illustrating the potential difficulty of the task. SPGym's design allows for unlimited scalability, including the potential for generating new image datasets on the fly using diffusion models, further increasing its utility as a benchmark for future research and development of more capable open-world agents.

Limitations. While SPGym presents a versatile framework, this work has several limitations. First, our experiments did not include agents specifically designed for representation learning, such as DARLA, CURL, RAD, DrQ, CBM, and CycAug. Future research should evaluate how SPGym ranks such agents in its challenging settings. Second, the high stochasticity of the environment and agents' sensitivity to the sampled image pools suggest that more seeds should be used in each experiment to ensure statistical robustness. However, this was impractical in our study due to the prohibitive computational costs.

Future Work. SPGym's adaptability opens several promising paths for future research. While this paper focuses on visual inputs, the benchmark supports extension to other data modalities, such as text, enabling investigation of how agents handle a variety of input types. Additionally, future work could explore in- and out-of-distribution generalization by training agents and encoders on specific image pools, external datasets, or through unsupervised learning methods, and testing their performance on unseen or novel data classes. Another valuable direction is the integration of curriculum learning into SPGym, where agents begin with simpler, partially solved puzzles and gradually face more difficult configurations. These extensions would deepen our understanding of how RL agents learn and generalize in complex, real-world environments.

Acknowledgments and Disclosure of Funding

The authors would like to thank Advanced Knowlegde Center for Immersive Technologies (AKCIT) for their support in conducting this study.

References

[1] Charles Beattie, Joel Z Leibo, Denis Teplyashin, Tom Ward, Marcus Wainwright, Heinrich Küttler, Andrew Lefrancq, Simon Green, Víctor Valdés, Amir Sadik, et al. Deepmind lab. *arXiv preprint arXiv:1612.03801*, 2016. 1, 2

[2] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013. 1, 2

[3] Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 2048–2056. PMLR, 13–18 Jul 2020. URL `https://proceedings.mlr.press/v119/cobbe20a.html`. 1, 2

[4] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023. 2, 5, 9

[5] Irina Higgins, Arka Pal, Andrei Rusu, Loic Matthey, Christopher Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. Darla: Improving zero-shot transfer in reinforcement learning. In *International Conference on Machine Learning*, pages 1480–1490. PMLR, 2017. 2

[6] Wm. Woolsey Johnson and William E. Story. Notes on the '15' puzzle. *American Journal of Mathematics*, 2(4):397–404, 1879. doi: 10.2307/2369492. URL `https://doi.org/10.2307/2369492`. Accessed: September 19, 2024. 4

[7] Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020. 2

[8] Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pages 5639–5650. PMLR, 2020. 2

[9] Misha Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. *Advances in neural information processing systems*, 33:19884–19895, 2020. 2

[10] Qiyuan Liu, Qi Zhou, Rui Yang, and Jie Wang. Robust representation learning by clustering with bisimulation metrics for visual reinforcement learning with distractions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8843–8851, 2023. 2

[11] Guozheng Ma, Linrui Zhang, Haoyu Wang, Lu Li, Zilin Wang, Zhen Wang, Li Shen, Xueqian Wang, and Dacheng Tao. Learning better with less: effective augmentation for sample-efficient visual reinforcement learning. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[12] Bruno Marotta. How to solve any slide puzzle regardless of its size. `https://kopf.com.br/kaplof/how-to-solve-any-slide-puzzle-regardless-of-its-size/`, 2017. Accessed: September 19, 2024. 4

[13] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y. 5

[14] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2, 5

[15] Austin Stone, Oscar Ramirez, Kurt Konolige, and Rico Jonschkowski. The distracting control suite–a challenging benchmark for reinforcement learning from pixels. *arXiv preprint arXiv:2101.02722*, 2021. 1, 2

[16] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018. 1, 2

[17] GuiPing Wang and Ren Li. Dsolving: a novel and efficient intelligent algorithm for large-scale sliding puzzles. *Journal of Experimental & Theoretical Artificial Intelligence*, 29(4):809–822, 2017. doi: 10.1080/0952813X.2016.1259270. URL `https://doi.org/10.1080/0952813X.2016.1259270`. 4

A Hyperparameters

Table 1 lists hyperparameters used across all experiments, unless noted otherwise. For DreamerV3, we adopted hyperparameters from [4], modifying only the decoder loss scale (set to 0) for the version without decoder.

Table 1: Hyperparameters for PPO and DreamerV3

Algorithm	Hyperparameter	Value
PPO	Env steps	10M
	Env instances	1024
	Optimizer	Adam
	Learning Rate (LR)	2.5e-4
	LR annealing	Yes
	Adam Epsilon	1e-5
	Num. steps	4
	Num. epochs	4
	Batch size	64
	Num. minibatches	4
	Gamma	0.99
	GAE lambda	0.95
	Advantage normalization	Yes
	Clip coef.	0.1
	Clip value loss	Yes
	Value function coef.	0.5
DreamerV3	Env steps	10M
	Env instances	16
	Model size	12M
	Replay capacity	5e6
	Replay ratio	32
	Action repeat	1
	Learning rate	4e-5
	Batch size	16
	Batch length	64
	Imagination horizon	15
	Discount horizon	333