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. 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, disentangling representation learning from other learning tasks, such as policy optimization or dynamics modeling, remains a key challenge in RL benchmarks.

Most RL benchmarks, including widely adopted ones like Atari [2], DeepMind Control Suite [16], and DeepMind Lab [1], entangle representation learning with other learning challenges, making it difficult to isolate the agent's ability to form robust representations. More specialized benchmarks, such as the Distracting Control Suite [15], emphasize robustness to visual noise but focuses on filtering out irrelevant complexity.

[*]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 scalable RL benchmark designed to test agents' visual representation learning capabilities. SPGym extends the 15-tile puzzle with variable grid sizes and observation spaces, such as image-based tiles, allowing for a controlled and scalable increase in representation complexity. Crucially, SPGym keeps the underlying environment dynamics and task structure constant, isolating the representation learning challenge.

We evaluate the performance of both model-free and model-based RL agents on SPGym, comparing their ability to generalize across increasingly complex visual input spaces. Our experiments reveal that, as representation learning complexity increases, SPGym effectively differentiates between agents based on their ability to form compositional and generalizable state representations. Specifically, model-based agents such as 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. SPGym achieves this by gradually increasing the complexity of visual observations without changing the underlying task structure.
- We provide an experimental evaluation of model-free and model-based RL agents, demonstrating the benchmark's effectiveness in distinguishing between agents' representation learning capabilities.
- We reveal limitations in current RL methods' ability to handle complex and diverse visual inputs, showcasing the scalability and challenge SPGym offers as a benchmark.

Our experiments show that SPGym offers a challenging testbed for visual representation learning in RL, making it a valuable tool for future research aimed at improving agents' visual representation capabilities in open-world environments.

2 Related Work

Reinforcement learning benchmarks are essential tools for evaluating agents' capabilities across diverse tasks. Popular benchmarks like Atari [2], DeepMind Control Suite [16], and DeepMind Lab [1] focus primarily on assessing policy learning and generalization. Yet, these benchmarks inherently combine representation learning with policy optimization and dynamics modeling, making it difficult to isolate the specific impact of representation learning on an agent's overall performance. Even methods aimed at improving 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. While these methods improve agents' ability to learn from raw visual inputs, they do not provide a clear, isolated assessment of their representation learning capabilities.

ProcGen [3] takes a step toward evaluating generalization through diversity by offering procedurally generated levels, challenging agents to adapt to unseen environments. However, like other benchmarks, it does not fully separate representation learning from policy and dynamics learning. Agents must simultaneously learn representations, policies, and environment dynamics, making it difficult to attribute performance improvements solely to better representations. 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.

SPGym's design sets it apart from existing benchmarks by providing a controlled environment where difficulty arises solely from the complexity of visual observations, rather than changes in task dynamics or objectives. This makes SPGym particularly well-suited for comparing RL algorithms that specifically target improvements in representation learning.

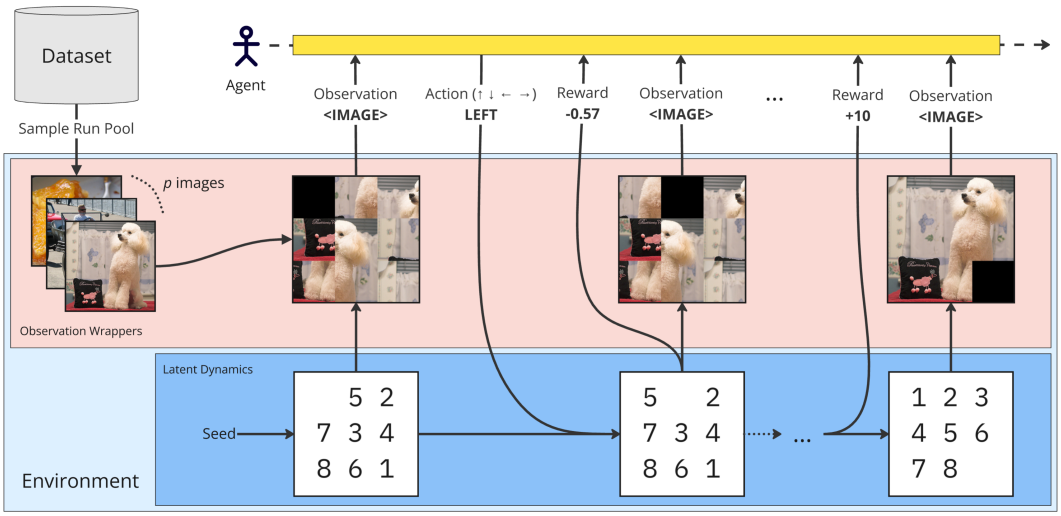

Figure 1: **Overview of the Sliding Puzzles Gym (SPGym).** The framework extends the 15-tile puzzle by incorporating image-based tiles, allowing scalable representation complexity while maintaining fixed environment dynamics.

3 The Sliding Puzzles Gym

The Sliding Puzzles Gym (SPGym) extends the 15-tile puzzle, where players 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, ranging from 2×2 to larger $H \times W$ configurations, and incorporating diverse observation spaces. In this work, we focus on 3×3 grids to emphasize the challenge of visual representation learning using image-based observations.

SPGym's internal representation is a 2D array of tile indices representing the current grid state. We provide wrappers that convert these states into agent-friendly observations, such as one-hot encodings or, as used in this paper, image overlays. Figure 1 presents the main components of SPGym.

At each time step, the reward is computed as

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

with

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

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.

SPGym provides two methods for generating the puzzle's initial state. The first method, used in this paper, generates a random $H \times W$ array and adjusts the parity of the puzzle by swapping the first two tiles if necessary to ensure solvability [6]. The second method, which can support curriculum

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

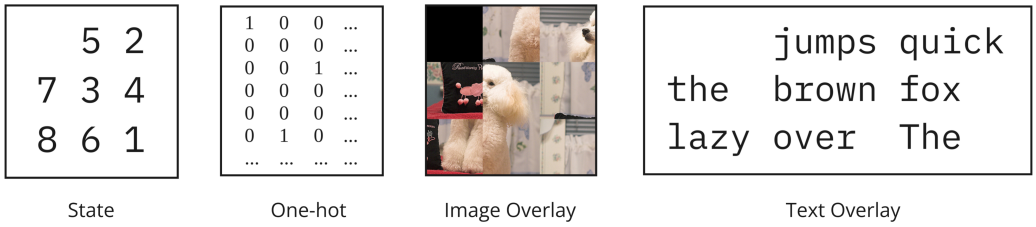

Figure 2: **Different observation modalities in SPGym.** Each modality presents a unique challenge for representation learning. The four presented observations represent the same latent puzzle state.

learning in future work, starts with a solved puzzle and applies a random series of valid moves to generate an initial state, enabling easier starting puzzle configurations.

Wrappers and Observation Modalities SPGym offers flexible wrappers that transform internal states into different modalities, ranging from simple one-hot encodings to complex visual observations, such as real-world images and text (see Figure 2). This flexibility makes SPGym adaptable for exploring how reinforcement learning agents handle diverse observation spaces, enhancing its utility as a benchmark for representation learning. In this paper, however, we focus on visual representation learning, leaving other modalities for future exploration.

For image overlays, we sample p images from a predefined dataset to form a pool for each run. The benchmark is dataset-agnostic, enabling the use of any image dataset, including images generated procedurally via diffusion. At the beginning of each episode, SPGym selects a random image from the run pool, splits it into $H \times W$ indexed patches, and overlays it onto the puzzle. The agent must then solve the puzzle by reconstructing the shuffled image, testing its ability to form compositional visual representations from raw pixel data.

Complexity Scalability A key feature of our benchmark is its scalable difficulty. SPGym can increase 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 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 to evaluate the efficacy of policy-learning algorithms in more challenging settings, we find that a 3×3 grid is sufficient to differentiate the performance of the algorithms tested in this study.

By maintaining fixed environment dynamics and task mechanics across different levels of difficulty, 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.

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.

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

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

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

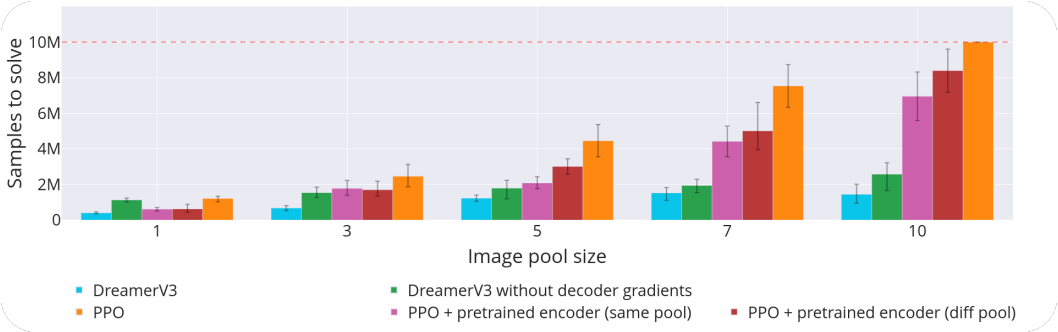

Figure 3: **Average number of steps to solve the puzzle as a function of the pool size.** We consider the puzzle solved when the agents reach 80% success rate and limit the maximum number of environment steps per run to 10 million. Error bars represent the 95% confidence interval of a bootstrap with 1,000 resamples from 5 independent seeds.

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, respectively, 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.

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]

5 Results

The results in Figure 3 demonstrate that as the complexity of the image pool increases, model-based agents, particularly the standard DreamerV3, consistently outperform their model-free counterparts. The DreamerV3 agent without the decoder performs noticeably worse, emphasizing the importance of DreamerV3's autoencoder structure in representation learning. For image pools with two or more images, DreamerV3 maintains a clear advantage over PPO-based agents, which is expected given the absence of dedicated representation learning mechanisms in PPO and the numerous optimizations present in DreamerV3.

Interestingly, we observe that PPO agents with pretrained encoders perform better than those without, even when the pretraining was done on different image pools. This suggests that these encoders capture transferable, task-agnostic features that aid performance. However, as the representation learning

[5]We attribute this runtime disparity primarily to DreamerV3's extensive utilization of swap space for its replay buffer.

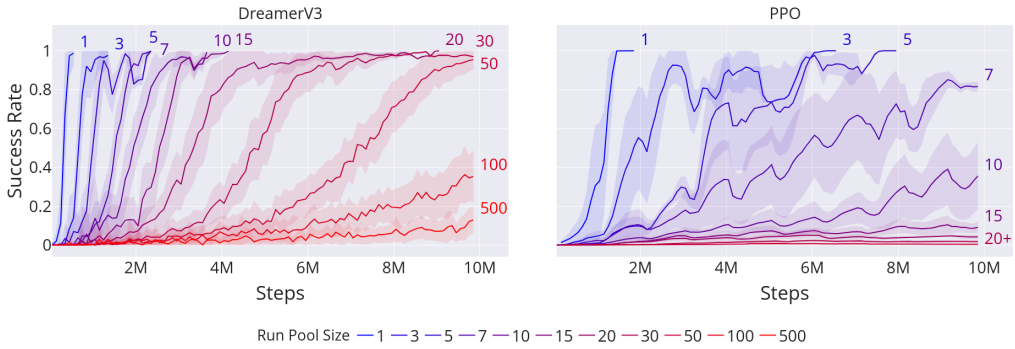

Figure 4: **Success rate as a function of environment steps.** The gradual increase in representation complexity affects the sample efficiency of standard PPO and DreamerV3 agents. Error bands indicate the 95% confidence interval, derived from 5 independent seeds and a rolling window of 100,000 environment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

challenge increases (i.e., larger image pools), the efficiency of all agents diminished. This trend reflects the scaling difficulty introduced by SPGym, further highlighting its utility as a benchmark for testing agents' ability to generalize across diverse visual inputs.

Figure 4 illustrates the success rates as a function of environment steps for both standard DreamerV3 and PPO agents, across various image pool sizes. As the pool size increases, both agents exhibit a decline in sample efficiency. Even the DreamerV3 agent, which generally performs well, fails to solve the puzzle when the pool size scales, underscoring the difficulty of scaling representation learning in complex visual environments. This result demonstrates the potential of SPGym to challenge even state-of-the-art agents, revealing limitations in their ability to generalize from raw visual inputs.

Overall, our experiments show that SPGym effectively discriminates between agents based on their ability to form robust and generalizable state representations. The increasing difficulty, driven by visual diversity, highlights key challenges that remain in representation learning for decision-making tasks. These results open up several avenues for future research, particularly in developing agents that can handle 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 outperformed 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) and the Center of Excellence in Artificial Ingelligence (CEIA) 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`. 2

[4] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023. 2, 4, 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. 3

[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. 4

[14] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2, 4

[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