DREAMER-CDP: IMPROVING RECONSTRUCTION-FREE WORLD MODELS VIA CONTINUOUS DETERMINISTIC REPRESENTATION PREDICTION [TINY PAPER]

Michael Hauri & Friedemann Zenke
Friedrich Miescher Instiute for Biomedical Research
Fabrikstrasse 24
4056 Basel, Switzerland
{michael.hauri,friedemann.zenke}@fmi.ch

ABSTRACT

Model-based reinforcement learning (MBRL) agents operating in high-dimensional observation spaces, such as Dreamer, rely on learning abstract representations for effective planning and control. Existing approaches typically employ reconstruction-based objectives in the observation space, which can render representations sensitive to task-irrelevant details. Recent alternatives trade reconstruction for auxiliary action prediction heads or view augmentation strategies, but perform worse in the Crafter environment than reconstruction-based methods. We close this gap between Dreamer and reconstruction-free models by introducing a JEPA-style predictor defined on continuous, deterministic representations. Our method matches Dreamer's performance on Crafter, demonstrating effective world model learning on this benchmark without reconstruction objectives.

1 INTRODUCTION

Recent progress in MBRL has enabled data-efficient learning in high-dimensional observation spaces. Central to these methods is learning a latent dynamics (or "world") model, which is subsequently used for downstream tasks such as planning, control, and policy optimization. To be effective, this world model must operate on abstract, compressed representations that capture task-relevant structure while discarding irrelevant details.

Self-supervised learning (SSL) has proven effective for learning such representations from experience. Classic works rely on reconstruction objectives (Ha & Schmidhuber, 2018; Hafner et al., 2019b). However, reconstruction may bias representations toward pixel-level details that are largely irrelevant for behavior (Nguyen et al., 2021; Zhang et al., 2024; Voelcker et al., 2024). This insight is generating interest in reconstruction-free SSL. To that end, several works studied reconstruction-free variants (Deng et al., 2022; Burchi & Timofte, 2024) of Dreamer, a widely-used MBRL framework (Hafner et al., 2019a). However, these strategies do not match the performance of reconstruction-based approaches on challenging benchmarks such as Crafter. This shortcoming may be a consequence of training both the representation and transition models to predict Dreamer's discrete probabilistic state variables.

We combine recent SSL ideas into Dreamer-CDP, which learns a world model by adding continuous deterministic representation prediction (CDP), while matching Dreamer's performance on Crafter.

2 PRIOR WORK

Preventing collapse in reconstruction-free SSL: One approach is the use of contrastive objective functions (Oord et al., 2018). However, contrastive methods often require large batch sizes (Chen et al., 2022), typically violate temporal locality, and can suffer from the curse of dimensionality (LeCun, 2022). Another strategy to avoid collapse leverages joint-embedding predictive architectures (JEPAs) (LeCun, 2022; Garrido et al., 2024), which trades contrastive objectives for additional

regularization techniques (Bardes et al., 2021), or predictor networks with specific stop-gradient placement such as BYOL (Grill et al., 2020) and SimSiam (Chen & He, 2021).

Reconstruction-free SSL in reinforcement learning (RL) has been explored in settings with high-dimensional input spaces, such as images. Zheng et al. (2023a;b); Burchi & Timofte (2025) used contrastive learning methods. Several other works employed self-predictive learning (Ni et al., 2024) either to learn directly state representations (Gelada et al., 2019; Schwarzer et al., 2020) or to train a world model for downstream control and planning tasks (Zhou et al., 2024; Sobal et al., 2025; Assran et al., 2025). Closely related to our work, BYOL-explore (Guo et al., 2022) introduced a parsimonious design based on Grill et al. (2020) to solve Atari games. In addition, EfficientZero (Ye et al., 2021; Wang et al., 2024) and TD-MPC2 (Hansen et al., 2023) adopted a SimSiam-style architecture. However, in contrast to Dreamer models, these algorithms rely on a purely non-stochastic continuous-variable world model.

Within the context of Dreamer, Okada & Taniguchi (2022) proposed to use contrastive-learning methods, whereas Deng et al. (2022) integrated prototypical representations (Caron et al., 2020) to temporal dynamics learning in DreamerPro. In MuDreamer, Burchi & Timofte (2024) proposed to use the action signal to train the world model (Table 1). Despite this diversity of existing approaches, reconstruction-based models remain the gold standard for Crafter (see Table 2).

Method	Reconstruction-free	Non-contrastive	No action prediction	No view augmentation
Dreamer	○	○	●	●
DreamerPro	●	●	●	○
MuDreamer	●	○	○	●
Dreamer-CDP	●	●	●	●

Table 1: Overview of Dreamer variants. Dreamer relies on pixel-based reconstruction. Other methods are reconstruction-free: MuDreamer uses action prediction to train the sequence model. DreamerPro utilizes augmented views. Dreamer-CDP (this article) relies solely on internal prediction.

3 THE DREAMER FRAMEWORK

Dreamer is an MBRL algorithm that learns a world model and uses imagined trajectories for policy learning, thereby instantiating the Dyna framework (Sutton, 1991) in a high-dimensional, pixel-based setting. Here, we briefly recap the DreamerV3 implementation (Hafner et al., 2025). The current observation x_t is encoded into a discrete stochastic state z_t. The sequence model predicts the next hidden state h_{t+1}, from h_t, z_t, and the action leading to the next state a_t. The dynamics are learned by reconstructing the next input $\hat{x}_{t+1}$. The model is summarized below:

$$\begin{aligned}
\text{Sequence model: } h_t &= f_\phi(h_{t-1}, z_{t-1}, a_{t-1}) \\
\text{Encoder: } z_t &\sim q_\phi(z_t|h_t, x_t) \\
\text{Dynamics predictor: } \hat{z}_t &\sim p_\phi(\hat{z}_t|h_t) \\
\text{Reward predictor: } \hat{r}_t &\sim p_\phi(\hat{r}_t|h_t, z_t) \\
\text{Continuation flag predictor: } \hat{c}_t &\sim p_\phi(\hat{c}_t|h_t, z_t) \\
\text{Decoder: } \hat{x}_t &\sim p_\phi(\hat{x}_t|h_t, z_t)
\end{aligned}$$

where c_t is the continuation flag and r_t is the reward at time t. The world model is trained with the following loss:

$$\mathcal{L}(\phi) = E_{q_\phi}\left[\sum_t \left(\beta_{\text{recon}}\mathcal{L}_{\text{recon}}(\phi) + \beta_{\text{aux}}\mathcal{L}_{\text{aux}}(\phi) + \beta_{\text{dyn}}\mathcal{L}_{\text{dyn}}(\phi) + \beta_{\text{rep}}\mathcal{L}_{\text{rep}}(\phi)\right)\right] \tag{1}$$

with

$$\begin{aligned}
\mathcal{L}_{\text{recon}}(\phi) &= -\ln p_\phi(x_t|z_t, h_t) \\
\mathcal{L}_{\text{aux}}(\phi) &= -\ln p_\phi(r_t|z_t, h_t) - \ln p_\phi(c_t|z_t, h_t) \\
\mathcal{L}_{\text{dyn}}(\phi) &= \max(1, \mathrm{D}_{\text{KL}}[\mathrm{SG}(q_\phi(z_t|h_t, x_t))||p_\phi(z_t|h_t)]) \\
\mathcal{L}_{\text{rep}}(\phi) &= \max(1, \mathrm{D}_{\text{KL}}[q_\phi(z_t|h_t, x_t)||\mathrm{SG}(p_\phi(z_t|h_t))]) \quad ,
\end{aligned}$$

where SG is the stop-grad operator and D_{KL} is the Kullback-Leibler (KL) divergence.

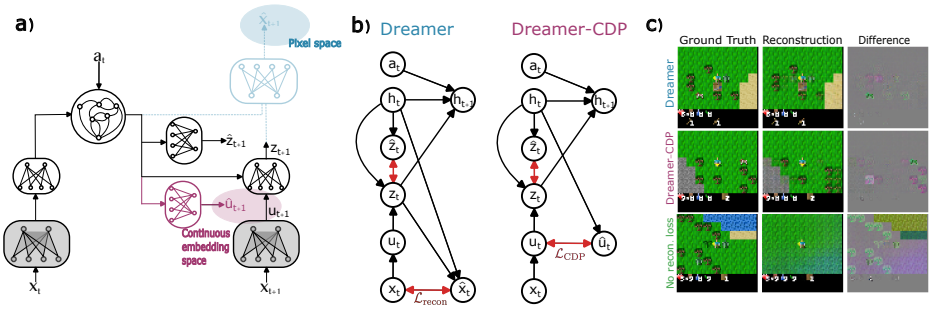

Figure 1: **a)** Schematic of Dreamer-CDP. The hidden state is passed through a predictor (green) trained to approximate the next continuous representation $\hat{u}_t \approx u_t$. In Dreamer, the hidden state and the input embedding are used to predict the next input x_t (dashed gray). **b)** Graphical model of Dreamer (left) and Dreamer-CDP (right) with losses in red. **c)** Visual examples when $\mathcal{L}_{\text{recon}}$ (Dreamer), $\mathcal{L}_{\text{CDP}}$ (Dreamer-CDP) or neither is applied. For the latter two, the decoder was trained independently with detached gradients for visualization purposes.

4 Reconstruction-free World Model Learning in Dreamer-CDP

To show how to learn efficient world models without reconstruction, we introduce Dreamer-CDP[1], a simple variant of DreamerV3 (Hafner et al., 2025) in which we remove the reconstruction loss while adding a JEPA-style predictor for CDP (Fig. 1a,b) inspired by recent work on temporal JEPAs (Mohammadi et al., 2025). To that end, we separate Dreamer's original representation encoder $q_\phi(z_t|h_t, x_t)$ as follows: First, observations x_t are mapped to a continuous deterministic embedding u_t via a feature extractor. A stochastic encoder then predicts a latent state representation $z_t \sim p_\phi(z_t|h_t, u_t)$ from the features u_t and the hidden state h_t. The latent state, together with the current action a_t, is processed by a recurrent dynamics model to yield h_{t+1}. While a feedforward architecture would be sufficient under full observability (Hansen et al., 2023), here we follow the Dreamer lineage in which the recurrent neural network (RNN) is essential to deal with partial observability. However, RNNs create distinct challenges for predictive SSL (Mohammadi et al., 2025). To navigate these challenges, we train the predictor $\hat{u}_{t+1} = g_\phi(h_{t+1})$ on the continuous deterministic embeddings u_{t+1} of the future observation x_{t+1}, which does not depend on the hidden state h_{t+1}. In contrast to Guo et al. (2022), we do not use an exponential moving average (EMA) target network. Instead, we rely on the insight that the sequence model must be close to a fixed point of its dynamics when the parameters of the representation network are updated (Tang et al., 2023; Khetarpal et al., 2025). To ensure this convergence, we train the sequence model predictors with a higher learning rate (see A.1) using the following objective:

$$\mathcal{L}(\phi) = E_{q_\phi}\left[\sum_t \left(\beta_{\text{CDP}}\mathcal{L}_{\text{CDP}}(\phi) + \beta_{\text{aux}}\mathcal{L}_{\text{aux}}(\phi) + \beta_{\text{dyn}}\mathcal{L}_{\text{dyn}}(\phi) + \beta_{\text{rep}}\mathcal{L}_{\text{rep}}(\phi)\right)\right] \tag{2}$$

where $\mathcal{L}_{\text{CDP}}$ is given by the negative cosine similarity $\mathcal{L}_{\text{CDP}}(\phi) = -\sum_t \cos(\text{SG}(u_{t+1}), \hat{u}_{t+1})$.

It is worth noting that the original Dreamer architecture already incorporates internal prediction through its KL balancing mechanisms (Hafner, 2021), which arises from the KL regularization term in the evidence lower bound (ELBO) objective (Hafner et al., 2019b). However, this prediction mechanism leverages the purely probabilistic discrete targets for learning (cf. Fig. 1b) and by itself does not lead to high-performing world models, as we will see below.

5 Experiments

To evaluate Dreamer-CDP, we used Crafter (Hafner, 2021), a computationally lightweight version of Minecraft, allowing us to assess agents on long-term reasoning, exploration, generalization, and dealing with sparse rewards. Performance was measured using the Crafter score, a metric that weighs the discovery of new achievements more strongly than the exploitation of already unlocked

[1]Code is available at: https://github.com/fmi-basel/Dreamer-CDP

ones. For instance, unlocking a new achievement that is reached in only 1% of episodes yields a larger score increase than improving the success rate of an existing achievement from 90% to 100%.

Implementation. We trained all models on a single Nvidia V100 GPU. The model comprised a MLP deterministic predictor and a Recurrent State-Space Model (RSSM) (Hafner et al., 2019b) with a CNN encoder (see A.1 for details). Each model interacted 1M times with the environment.

Baseline methods. We compared Dreamer-CDP to three different algorithms (Table 1). First, the original DreamerV3 (Hafner et al., 2025), which learns the world model by reconstruction in the input space (Sec. 3). We also compared it to MuDreamer (Burchi & Timofte, 2024), which, inspired by MuZero (Schrittwieser et al., 2020), trains the world model by predicting the action $\hat{a}_t \sim p_\phi(\hat{a}_t|h_t, z_t, x_{t+1})$ that lead to the current state and the value $\hat{v}_t \sim p_\phi(\hat{v}_t|h_t, z_t)$. MuDreamer attained performance comparable to Dreamer on the Atari and DeepMind Control benchmark, even outperforming it when trained with naturalistic backgrounds. Finally, we also compared it to DreamerPro (Deng et al., 2022), another non-contrastive SSL method, which combines prototypical representations (Caron et al., 2020) with learning a sequence model, but by predicting jointly the cluster assignment of the observation and an augmented view rather than predicting the next latent state.

5.1 Results

Dreamer-CDP achieved a Crafter score of $16.2 \pm 2.1\%$ (Table 2; Fig. A.1) on par with DreamerV3 ($14.5 \pm 1.6\%$; t-test $p = 0.10$) and only outperformed by introducing prioritized experience replay ($19.4 \pm 1.6\%$; Kauvar et al., 2023). To check that these results did indeed depend on the prediction of deterministic target embeddings, we trained the same model without $\mathcal{L}_{\text{CDP}}$, which is equivalent to classical Dreamer without $\mathcal{L}_{\text{recon}}$. This manipulation resulted in an expected performance drop ($3.2 \pm 1.2\%$; Fig. A.2). We next compared Dreamer-CDP to other approaches. MuDreamer exhibited a notably lower Crafter score of $7.3 \pm 2.6\%$ (Table 2). This performance gap can likely be attributed to the comparatively weak action signal in Crafter. While we did not train DreamerPro, Kauvar et al. (2023) reported a Crafter score of $4.7 \pm 0.5\%$. Thus, Dreamer-CDP performs on par with Dreamer and better than previous reconstruction-free approaches.

To check to what extent reward prediction contributed to Dreamer-CDP's performance, we retrained the model without propagating gradients from the reward predictor head. We found that this ablation resulted in an intermediate performance drop to $12.7 \pm 1.6\%$ (Fig. A.2). In contrast, when training Dreamer-CDP without the alignment objectives $\mathcal{L}_{\text{dyn/rep}}$, performance decreased to $6.3 \pm 1.9\%$. Thus CDP is necessary but not sufficient to improve reconstruction-free world models.

Metrics	Dreamer	DreamerPro	MuDreamer	Dreamer-CDP (ours)
Crafter score	$14.5 \pm 1.6\%^\dagger$	$4.7 \pm 0.5\%^\dagger$	$7.3 \pm 2.6\%$	$16.2 \pm 2.1\%$
Cum. reward	$11.7 \pm 1.9^\dagger$	—	5.6 ± 1.6	9.8 ± 0.4

Table 2: Crafter score and cum. reward $\pm$ std ($n = 7$) of different Dreamer variants (cf. Sec. 5). † published results by Hafner et al. (2023) and Kauvar et al. (2023).

6 Conclusion

In this work, we showed that including CDP is essential for learning a reconstruction-free world model that matches the reconstruction-based Dreamer reference implementation on Crafter. An important direction for future research is to identify and benchmark other environments in which predictive learning provides advantages. On the one hand, we expect computational savings owed to removing the decoder in complex environments. On the other hand, we believe that reconstruction-free world models open the door to improved data-efficiency in complex high-dimensional environments with simple action signals and sparse reward structure.

Acknowledgments

The authors thank Ashena Gorgan Mohammadi, Peter Buttaroni, Fabian Mikulasch, and all members of the Zenke Lab for their thoughtful input. This project was supported by the Swiss National

Science Foundation (Grant Number PCEFP3_202981), EU's Horizon Europe Research and Innovation Programme (Grant Agreement No. 101070374, CONVOLVE) funded through SERI (Ref. 1131– 52302), and the Novartis Research Foundation.

REFERENCES

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Maxime Burchi and Radu Timofte. Mudreamer: Learning predictive world models without reconstruction. *arXiv preprint arXiv:2405.15083*, 2024.

Maxime Burchi and Radu Timofte. Learning transformer-based world models with contrastive predictive coding. *arXiv preprint arXiv:2503.04416*, 2025.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

Changyou Chen, Jianyi Zhang, Yi Xu, Liqun Chen, Jiali Duan, Yiran Chen, Son Tran, Belinda Zeng, and Trishul Chilimbi. Why do we need large batchsizes in contrastive learning? a gradient-bias perspective. *Advances in Neural Information Processing Systems*, 35:33860–33875, 2022.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15750–15758, 2021.

Fei Deng, Ingook Jang, and Sungjin Ahn. Dreamerpro: Reconstruction-free model-based reinforcement learning with prototypical representations. In *International conference on machine learning*, pp. 4956–4975. PMLR, 2022.

Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning. *arXiv preprint arXiv:2403.00504*, 2024.

Carles Gelada, Saurabh Kumar, Jacob Buckman, Ofir Nachum, and Marc G Bellemare. Deepmdp: Learning continuous latent space models for representation learning. In *International conference on machine learning*, pp. 2170–2179. PMLR, 2019.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

Zhaohan Guo, Shantanu Thakoor, Miruna Pîslar, Bernardo Avila Pires, Florent Altché, Corentin Tallec, Alaa Saade, Daniele Calandriello, Jean-Bastien Grill, Yunhao Tang, et al. Byol-explore: Exploration by bootstrapped prediction. *Advances in neural information processing systems*, 35: 31855–31870, 2022.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3), 2018.

Danijar Hafner. Benchmarking the spectrum of agent capabilities. *arXiv preprint arXiv:2109.06780*, 2021.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019b.

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

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, pp. 1–7, 2025.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. *arXiv preprint arXiv:2310.16828*, 2023.

Isaac Kauvar, Chris Doyle, Linqi Zhou, and Nick Haber. Curious replay for model-based adaptation. *arXiv preprint arXiv:2306.15934*, 2023.

Khimya Khetarpal, Zhaohan Daniel Guo, Bernardo Avila Pires, Yunhao Tang, Clare Lyle, Mark Rowland, Nicolas Heess, Diana L Borsa, Arthur Guez, and Will Dabney. A unifying framework for action-conditional self-predictive reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 181–189. PMLR, 2025.

Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

Ashena Gorgan Mohammadi, Manu Srinath Halvagal, and Friedemann Zenke. Understanding cortical computation through the lens of joint-embedding predictive architectures. *bioRxiv*, pp. 2025–11, 2025.

Tung D Nguyen, Rui Shu, Tuan Pham, Hung Bui, and Stefano Ermon. Temporal predictive coding for model-based planning in latent space. In *International Conference on Machine Learning*, pp. 8130–8139. PMLR, 2021.

Tianwei Ni, Benjamin Eysenbach, Erfan Seyedsalehi, Michel Ma, Clement Gehring, Aditya Mahajan, and Pierre-Luc Bacon. Bridging state and history representations: Understanding self-predictive rl. *arXiv preprint arXiv:2401.08898*, 2024.

Masashi Okada and Tadahiro Taniguchi. Dreamingv2: Reinforcement learning with discrete world models without reconstruction. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 985–991. IEEE, 2022.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. *arXiv preprint arXiv:2007.05929*, 2020.

Vlad Sobal, Wancong Zhang, Kyunghyun Cho, Randall Balestriero, Tim GJ Rudner, and Yann LeCun. Learning from reward-free offline data: A case for planning with latent dynamics models. *arXiv preprint arXiv:2502.14819*, 2025.

Richard S Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4):160–163, 1991.

Yunhao Tang, Zhaohan Daniel Guo, Pierre Harvey Richemond, Bernardo Avila Pires, Yash Chandak, Rémi Munos, Mark Rowland, Mohammad Gheshlaghi Azar, Charline Le Lan, Clare Lyle, et al. Understanding self-predictive learning for reinforcement learning. In *International Conference on Machine Learning*, pp. 33632–33656. PMLR, 2023.

Claas Voelcker, Tyler Kastner, Igor Gilitschenski, and Amir-massoud Farahmand. When does self-prediction help? understanding auxiliary tasks in reinforcement learning. *arXiv preprint arXiv:2406.17718*, 2024.

Shengjie Wang, Shaohuai Liu, Weirui Ye, Jiacheng You, and Yang Gao. Efficientzero v2: Mastering discrete and continuous control with limited data. *arXiv preprint arXiv:2403.00564*, 2024.

Weirui Ye, Shaohuai Liu, Thanard Kurutach, Pieter Abbeel, and Yang Gao. Mastering atari games with limited data. *Advances in neural information processing systems*, 34:25476–25488, 2021.

Di Zhang, Bowen Lv, Hai Zhang, Feifan Yang, Junqiao Zhao, Hang Yu, Chang Huang, Hongtu Zhou, Chen Ye, et al. Focus on what matters: Separated models for visual-based rl generalization. *Advances in Neural Information Processing Systems*, 37:116960–116986, 2024.

Chongyi Zheng, Ruslan Salakhutdinov, and Benjamin Eysenbach. Contrastive difference predictive coding. *arXiv preprint arXiv:2310.20141*, 2023a.

Ruijie Zheng, Xiyao Wang, Yanchao Sun, Shuang Ma, Jieyu Zhao, Huazhe Xu, Hal Daumé III, and Furong Huang. Taco: Temporal latent action-driven contrastive loss for visual reinforcement learning. *Advances in Neural Information Processing Systems*, 36:48203–48225, 2023b.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

A Appendix

A.1 Hyperparameters

We used the default parameters of the DreamerV3 XL model. $\beta_{\text{CDP}} = 500$. The predictor was a two-layer MLP with 8192 input units, 4096 hidden units, and 4096 output units. The training ratio was 32 instead of 512. To ensure stable learning dynamics, the learning rate of the RSSM, and the predictor was trained with a higher learning rate of $4 \cdot 10^{-4}$ compared to the encoder ($\text{lr} = 6 \cdot 10^{-6}$). All other networks were trained with a learning rate of $4 \cdot 10^{-5}$.

A.2 Supplementary Figures

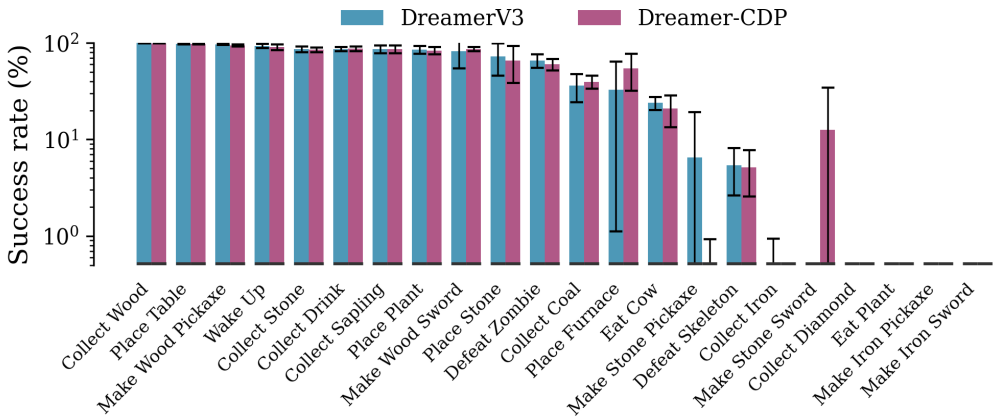

Figure A.1: Achievement success rate for DreamerV3 (Blue) and Dreamer-CDP (purple) sorted by DreamerV3 success rates.

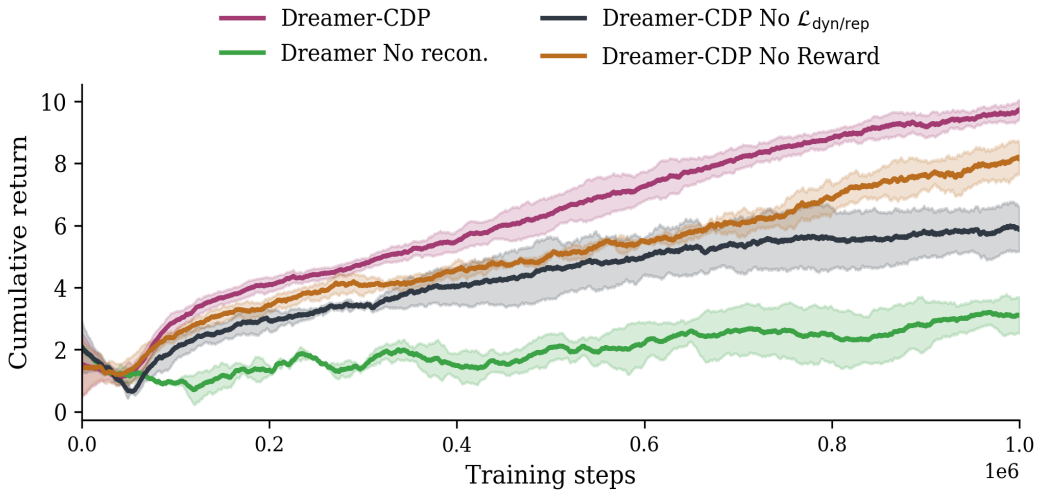

Figure A.2: Comparison between Dreamer-CDP (Purple) and several ablations. Orange: Ablation of the reward and value gradient to train the world model. Most of the learning signal comes from the latent space predictive loss. Green: Ablation of $\mathcal{L}_{\text{CDP}}$ and $\mathcal{L}_{\text{dyn/rep}}$. The KL balancing is not sufficient to train the world model in the latent space. Black: CDP loss alone also results in lower cumulative return.