Towards Efficient World Models

Eloi Alonso [*1] **Vincent Micheli** [*1] **François Fleuret** [1]

Abstract

Scaling up deep Reinforcement Learning (RL) agents beyond traditional benchmarks, without abundant computational resources, presents a significant challenge. Following recent developments in generative modelling, model-based RL positions itself as a strong contender to bring autonomous agents to new heights. In fact, the recently introduced IRIS agent provides evidence that advances in sequence modelling can be leveraged to build powerful world models. In the present work, we propose Δ-IRIS, a new agent with a world model architecture that is amenable to scaling up to visually complex environments with longer time horizons. In the Crafter benchmark, Δ-IRIS solves 16 out of 22 tasks after 10M frames of training, matching the current best method, DreamerV3.

1. Introduction

Deep Reinforcement Learning (DRL) methods have recently delivered impressive results (Ye et al., 2021; Hafner et al., 2023) in traditional benchmarks (Bellemare et al., 2013b), all while running on consumer-grade hardware. In light of the evermore complex domains tackled by the latest generations of generative models (OpenAI, 2023; Rombach et al., 2022), the prospect of training agents in more ambitious environments (Kanervisto et al., 2022) may hold significant appeal. However, that leap forward poses a serious challenge: DRL architectures have been comparatively smaller and less sample-efficient than their (self-)supervised counterparts. In contrast, more intricate environments necessitate models with greater representational power and have higher data requirements.

Model-based RL (MBRL) (Sutton & Barto, 2018) is hypothesized to be the key for scaling up DRL agents (LeCun, 2022). Indeed, world models (Ha & Schmidhuber, 2018) offer a diverse range of capabilities: lookahead search (Schrittwieser et al., 2020; Ye et al., 2021), learning in imagination (Kaiser et al., 2020; Hafner et al., 2023; Micheli et al., 2023), representation learning (D'Oro et al., 2023), and uncertainty estimation (Pathak et al., 2017; Sekar et al., 2020). In essence, MBRL shifts the focus from the RL problem to a generative modelling problem, where the development of an accurate world model significantly simplifies policy training. In particular, policies learnt in the imagination of world models are freed from sample efficiency constraints, a common limitation of RL agents that is magnified in complex environments with slow rollouts.

Recently, the IRIS agent (Micheli et al., 2023) delivered strong results in the Atari 100k benchmark (Bellemare et al., 2013b; Kaiser et al., 2020). IRIS introduced a world model composed of a discrete autoencoder and an autoregressive Transformer, casting dynamics learning as a sequence modelling problem where the Transformer composes over time a vocabulary of image tokens built by the autoencoder. Crucially, this approach opens up avenues for future methods to capitalize on advances in generative modelling (OpenAI, 2023; Villegas et al., 2022). However, in its current form, scaling IRIS to more complex environments is computationally prohibitive. Indeed, such an endeavor would require a large number of tokens to encode visually challenging frames. Besides, sophisticated dynamics may require to store numerous timesteps in memory to reason about the past, ultimately making the imagination procedure excessively slow. Hence, under these constraints, maintaining a favorable imagined-to-collected data ratio would be infeasible.

In the present work, we introduce Δ-IRIS, an improved version of the IRIS agent capable of scaling to visually complex environments with longer time horizons. Δ-IRIS encodes new frames by attending to the ongoing trajectory, effectively describing deltas between timesteps. This new approach drastically reduces the number of tokens to encode frames, since they are not encoded independently as in IRIS. In the Crafter benchmark (Hafner, 2022), Δ-IRIS unlocks 16 out of 22 objectives at the 10M frames mark, matching the performance of the current best agent, DreamerV3 (Hafner et al., 2023).

[*]Equal contributions, order determined by a coin flip.
[1]University of Geneva, Switzerland.
Correspondence to: {first.last}@unige.ch

Work presented at the ES-FoMo Workshop at ICML 2023.

2. Method

We consider a Partially Observable Markov Decision Process (POMDP) (Sutton & Barto, 2018). The environment dynamics is captured by the conditional distribution $p(x_{t+1}, r_t, d_t \mid x_{\leq t}, a_{\leq t})$, where $x_t \in \mathcal{X} = \mathbb{R}^{3\times h\times w}$ is an image observation, $a_t \in \mathcal{A} = \{1, \dots, A\}$ a discrete action, $r_t \in \mathbb{R}$ a scalar reward, and $d_t \in \{0, 1\}$ indicates episode termination. The reinforcement learning objective is to find a policy $\pi : \mathcal{X} \to \mathcal{A}$ that maximizes the expected sum of rewards $\mathbb{E}_\pi[\sum_{t\geq 0} \gamma^t r_t]$, with discount factor $\gamma \in (0, 1)$.

In the vein of IRIS (Micheli et al., 2023), our world model is composed of a discrete autoencoder and an autoregressive dynamics model. The reinforcement learning agent follows the one proposed in IRIS.

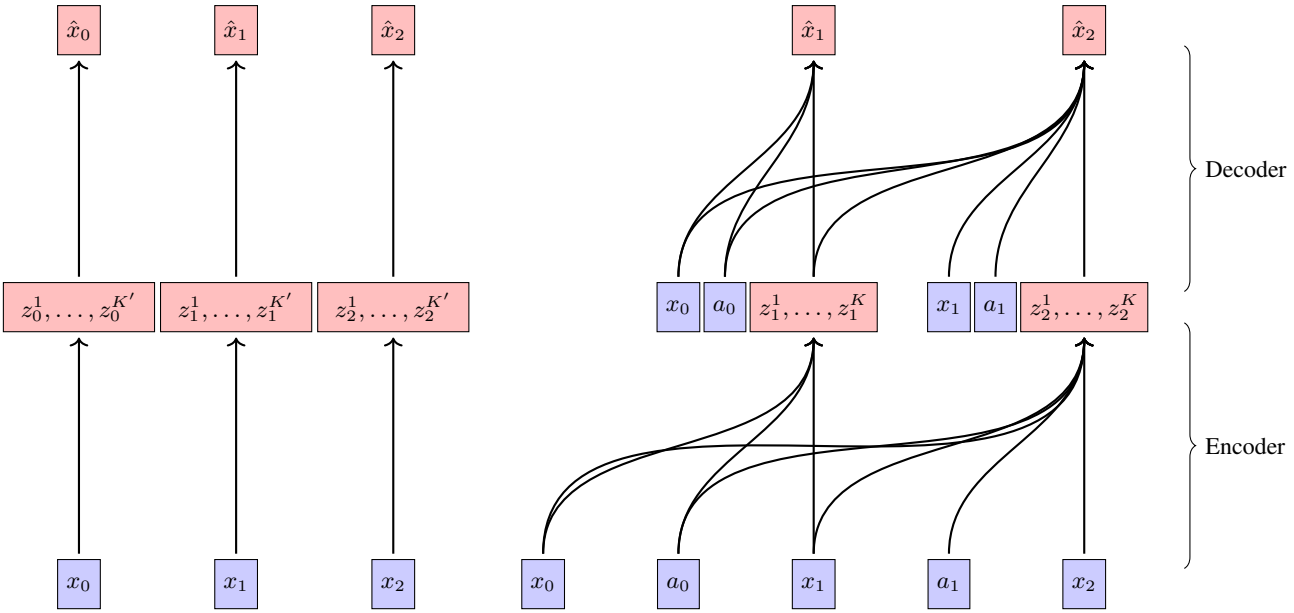

Figure 1. Discrete autoencoder of IRIS (left) and Δ-IRIS (right). IRIS encodes and decodes frames independently, meaning that the discrete tokens have to carry all the information to reconstruct each frame. Instead, Δ-IRIS's encoder and decoder are conditioned on past frames and actions, meaning that z_i just has to code for what changed with respect to past frames and that cannot be inferred from actions. This approach enables to drastically reduce the number of tokens to encode a frame in Δ-IRIS , which is critical to scale up imagination.

2.1. Discrete autoencoder

The discrete autoencoder (E, D) learns to represent high-dimensional images as a small number of discrete tokens from a vocabulary $\mathcal{Z} = \{1, \dots, N\}$. Compared to IRIS, the autoencoder is conditioned on the ongoing trajectory, allowing to drastically reduce the number of tokens K needed to represent a frame.

For any set $\mathcal{Y}$, we denote $\mathcal{S}_n(\mathcal{Y}) = \bigcup_{i=1}^n \mathcal{Y}^i$ the set of tuples of elements from $\mathcal{Y}$ of maximum length n, and $\mathcal{S}(\mathcal{Y}) = \mathcal{S}_\infty(\mathcal{Y})$.

Given past images and actions $(x_0, a_0, \dots, x_{t-1}, a_{t-1})$, the encoder $E : \mathcal{S}(\mathcal{X} \times \mathcal{A}) \times \mathcal{X} \to \mathcal{Z}^K$ converts an image x_t into $z_t = (z_t^1, \dots, z_t^K)$, a sequence of K discrete tokens, that we call transition tokens from now on. The encoder is parameterized by a stack of Transformer encoder layers with causal self-attention (Radford et al., 2019). Frames are sliced in non-overlapping patches (Dosovitskiy et al., 2021), and actions are embedded with a learnt lookup table. We interleave average pooling layers (Dai et al., 2020) to reduce the size of the sequence down to the number of desired transition tokens. This has the effect of decoupling the number of patches used to represent a frame and the number of tokens to encode it. We use vector quantization (Van Den Oord et al., 2017; Esser et al., 2021) with factorized and normalized codes (Yu et al., 2021) to discretize the encoder's continuous outputs.

While it should be possible to reconstruct an image, given a starting image, actions and transition tokens, we found it much more effective to incorporate previous frames in the input sequence of the decoder. Indeed, one would otherwise have to integrate over the transition tokens to reconstruct the current image, which is way harder. Hence, the decoder $D : \mathcal{S}(\mathcal{X} \times \mathcal{A} \times \mathcal{Z}^K) \to \mathcal{X}$ reconstructs an

image $\hat{x}_t$ from past frames, actions and transition tokens $(x_0, a_0, z_1, \ldots, x_{t-1}, a_{t-1}, z_t)$. The decoder is parameterized by a stack of Transformer encoder layers, where actions and transition tokens are embedded with learnt lookup tables, and frames are encoded as single continuous vectors with a convolutional neural network (CNN, LeCun et al., 1989). Reconstructed images are obtained by reassembling Transformer's outputs as patches. Again, to have more patches than transition tokens, we interleave upsampling layers between some Transformer layers, where transition tokens are duplicated and receive different positional embeddings.

The discrete autoencoder is trained on previously collected trajectories, with a weighted combination of L_1, L_2 and max-pixel (Anand et al., 2022) reconstruction losses, and a commitment loss (Van Den Oord et al., 2017). The codebook is updated with an exponential moving average (Razavi et al., 2019), and we revive unused codewords (Dhariwal et al., 2020; Zeghidour et al., 2021).

2.2. Dynamics model

The dynamics model G predicts next transition tokens, rewards, and episode ends. It is parameterized by a stack of Transformer encoder layers with causal self-attention. Similarly to the decoder D described above, we found it essential to incorporate frames in its input sequence.

Transition tokens are autoregressively predicted by the dynamics model. Given past frames, actions, and transition tokens $(x_0, a_0, z_1^1, \ldots, z_1^K, \ldots, x_{t-1}, a_{t-1}, z_t^1, \ldots, z_t^k)$, with $k \in \{0, \ldots, K-1\}$, the dynamics model outputs a categorical distribution on $\mathcal{Z}$ for the next transition token z_t^{k+1}. Target transition tokens are obtained with the discrete autoencoder from trajectories collected by the agent in the real environment. We follow DreamerV3 in using discrete regression with two-hot targets and symlog scaling for rewards prediction (Bellemare et al., 2017; Imani & White, 2018).

3. Experiments

Recent advances have started to show the limitations of well established benchmarks. For instance, in Atari 100k (Bellemare et al., 2013a; Kaiser et al., 2020), agents (Ye et al., 2021; Micheli et al., 2023; D'Oro et al., 2023; Hafner et al., 2023) now outperform humans with as little as two hours of training data available. Therefore, in the effort to build the next generation of agents capable of solving the most difficult and compute-hungry environments (Kanervisto et al., 2022), the following question arises: what benchmarks offer fast iteration cycles while reflecting some important mechanics found in advanced environments?

3.1. Benchmark and baselines

Crafter (Hafner, 2022) is a procedurally generated environment, inspired by the video game Minecraft, with visual inputs, a discrete action space and non-deterministic dynamics. By incorporating mechanics from survival games and a technology tree, this benchmark evaluates a wide range of agent abilities such as generalization, exploration, and credit assignment.

Two generations of Dreamer agents (Hafner et al., 2021; 2023) were evaluated on Crafter. DreamerV2 learns in the imagination of a world model combining a convolutional autoencoder with a recurrent state-space model (RSSM) (Hafner et al., 2019). The key modifications that enabled DreamerV2 to improve over the original Dreamer agent (Hafner et al., 2020) were categorical latents and KL balancing between prior and posterior estimates. DreamerV3 builds upon DreamerV2 with many additions such as symlog scaling of rewards and values, combining free bits (Kingma et al., 2016) with KL balancing, return scaling for fixed entropy regularization, and architectural novelties for model scaling.

IRIS is not featured as a baseline for computational reasons. Indeed, we observed in preliminary experiments that, in Crafter, IRIS requires 64 tokens to encode frames without losing too much information. In comparison, Δ-IRIS only requires 4 transition tokens per frame. Based on the training time for Δ-IRIS, we estimate that IRIS would take roughly 112 days of training to match the performance of Δ-IRIS.

3.2. Results

Table 1 displays returns at 10M frames, world model sizes, imagined-to-collected data ratios, and wall-clock times. Δ-IRIS was evaluated by computing an average over 256 episodes collected at the end of training. We ran our experiments with Nvidia A100 40GB GPUs.

After 10M frames of training, Δ-IRIS unlocks on average 16 objectives from the Crafter environment, matching the performance of the current best agent, DreamerV3. In addition, its world model and training ratio are smaller than DreamerV3's.

Table 1. Return at 10M frames, world model size, imagined-to-collected data ratio, and wall-clock time for Δ-IRIS and DreamerV3. Numbers for DreamerV3 are inferred from Figure 6a, Appendix A, and Appendix B in Hafner et al., 2023. With smaller world models and data ratios, Δ-IRIS matches the performance of DreamerV3.

Game	Return	WM size	Data ratio	Wall-clock time
Δ-IRIS	15.4	110M	24	7 days
DreamerV3	∼15.2	200M	64	∼2 days

4. Conclusion

We introduced Δ-IRIS , a new model-based agent relying on a discrete autoencoder and a dynamics model to simulate its environment. The key improvement over IRIS is to allow the discrete autoencoder to attend to the ongoing trajectory, effectively describing deltas between timesteps. We demonstrate experimentally on Crafter that Δ-IRIS matches the performance of DreamerV3, the current best agent on this benchmark.

Acknowledgements

Vincent Micheli was supported by the Swiss National Science Foundation under grant number FNS-187494.

References

Anand, A., Walker, J. C., Li, Y., Vértes, E., Schrittwieser, J., Ozair, S., Weber, T., and Hamrick, J. B. Procedural generalization by planning with self-supervised world models. In *International Conference on Learning Representations*, 2022.

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013a.

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013b.

Bellemare, M. G., Dabney, W., and Munos, R. A distributional perspective on reinforcement learning. In *International conference on machine learning*, pp. 449–458. PMLR, 2017.

Dai, Z., Lai, G., Yang, Y., and Le, Q. Funnel-transformer: Filtering out sequential redundancy for efficient language processing. *Advances in neural information processing systems*, 33:4271–4282, 2020.

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

D'Oro, P., Schwarzer, M., Nikishin, E., Bacon, P.-L., Bellemare, M. G., and Courville, A. Sample-efficient reinforcement learning by breaking the replay ratio barrier. In *The Eleventh International Conference on Learning Representations*, 2023.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12873–12883, 2021.

Ha, D. and Schmidhuber, J. Recurrent world models facilitate policy evolution. *Advances in neural information processing systems*, 31, 2018.

Hafner, D. Benchmarking the spectrum of agent capabilities. In *International Conference on Learning Representations*, 2022.

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., and Davidson, J. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020.

Hafner, D., Lillicrap, T. P., Norouzi, M., and Ba, J. Mastering atari with discrete world models. In *International Conference on Learning Representations*, 2021.

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

Imani, E. and White, M. Improving regression performance with distributional losses. In *International conference on machine learning*, pp. 2157–2166. PMLR, 2018.

Kaiser, Ł., Babaeizadeh, M., Miłos, P., Osiński, B., Campbell, R. H., Czechowski, K., Erhan, D., Finn, C., Kozakowski, P., Levine, S., et al. Model based reinforcement learning for atari. In *International Conference on Learning Representations*, 2020.

Kanervisto, A., Milani, S., Ramanauskas, K., Topin, N., Lin, Z., Li, J., Shi, J., Ye, D., Fu, Q., Yang, W., Hong, W., Huang, Z., Chen, H., Zeng, G., Lin, Y., Micheli, V., Alonso, E., Fleuret, F., Nikulin, A., Belousov, Y., Svidchenko, O., and Shpilman, A. Minerl diamond 2021 competition: Overview, results, and lessons learned. In *Proceedings of the NeurIPS 2021 Competitions and Demonstrations Track*, Proceedings of Machine Learning Research, 2022. URL https://proceedings.mlr.press/v176/kanervisto22a.html.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

LeCun, Y. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62, 2022.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

Micheli, V., Alonso, E., and Fleuret, F. Transformers are sample-efficient world models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=vhFu1Acb0xb.

OpenAI. Gpt-4 technical report, 2023.

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

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners, 2019.

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

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., Lillicrap, T. P., and Silver, D. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., and Pathak, D. Planning to explore via self-supervised world models. In *International Conference on Machine Learning*, pp. 8583–8592. PMLR, 2020.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Villegas, R., Babaeizadeh, M., Kindermans, P.-J., Moraldo, H., Zhang, H., Saffar, M. T., Castro, S., Kunze, J., and Erhan, D. Phenaki: Variable length video generation from open domain textual description. *arXiv preprint arXiv:2210.02399*, 2022.

Ye, W., Liu, S., Kurutach, T., Abbeel, P., and Gao, Y. Mastering atari games with limited data. *Advances in neural information processing systems*, 34, 2021.

Yu, J., Li, X., Koh, J. Y., Zhang, H., Pang, R., Qin, J., Ku, A., Xu, Y., Baldridge, J., and Wu, Y. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

Zeghidour, N., Luebs, A., Omran, A., Skoglund, J., and Tagliasacchi, M. Soundstream: An end-to-end neural audio codec, 2021.