Do Pre-Trained and Fine-Tuned World Models Generalize?

Georgy Savva

Abstract

This work investigates the generalization capabilities of two diffusion-based world models, OASIS and WorldMem within the Minecraft environment. OASIS is trained from scratch on the diverse VPT dataset, while WorldMem is a fine-tuned version of OASIS on a simpler, randomly generated dataset. The models are evaluated on three distinct datasets, VPT, WorldMem, and a custom-designed Consistency dataset, each representing a different distribution of the environment. Quantitative analysis using PSNR scores and qualitative video comparisons show that both models struggle to generalize beyond their training distributions, with fine-tuning also leading to catastrophic forgetting of the pretrained distribution. These findings reveal the limitations of the current world models in adapting to varied distributions and suggest that combining datasets for fine-tuning is necessary to preserve and extend the model's performance.

1. Introduction

Learning world models has been of significant interest to the reinforcement learning (RL) community (Hafner et al., 2020). By being an accurate approximation of a complex environment, they allow testing and evaluating agents. More importantly, by being fully differentiable, they enable end-to-end training and gradient-based methods to optimize the agent's actions, facilitating planning and looking ahead, which promises better agent performance in risk-critical environments like autonomous driving or robotics. A key innovation (Ha & Schmidhuber, 2018) was showing that agents can be trained entirely in "dream" environments generated by their world models, with policies successfully transferring to actual environments.

Video games are virtual worlds, and all the challenges and benefits of world models apply there as well. The Minecraft game has been used as the environment for agent training and exploration (Wang et al., 2023; Baker et al., 2022), and recently has emerged as the de facto platform for world model development due to its open-endedness and high complexity. The premise is that the world model should learn the game mechanics of the world and all the physics laws. Recent years have seen several attempts to make this work by employing video diffusion models (Ho et al., 2022). The two most recent notable works are OASIS (Decart & Julian Quevedo, 2024) and WorldMem (Xiao et al., 2025), with both models showing great performance in their demo videos.

A robust world model should learn the entire distribution of the world, and in this work, we study how the two publicly available models compare in this regard. Does their performance hold up on a test distribution different from the one they were trained on?

2. Related Works

2.1. Video Game Generation

To simulate a video game, the generative model needs to learn to predict the next observation based on the history of observations and actions. Formally, the model learns the following probability distribution:

$$p_\theta(o_t \mid o_{t-1}, a_{t-1} \ldots, o_0, a_0)$$

where o_t represents the observation at time t, and a_t represents the action at time t.

World models have been introduced as a way to simulate game environments through a combination of a variational auto-encoder (VAE) to learn over input frames (vision capabilities) and a recurrent neural network (RNN) to predict future distributions given past information (memory capabilities). This has been tested and shown to be effective in the VizDoom environment (Ha & Schmidhuber, 2018). Generative models are able to simulate games as demonstrated by GameGAN (Kim et al., 2020), which uses a combination of LSTM and GAN. It has been proposed that world models can be learned in latent space, since compact representations of the game state may improve the efficiency of planning and simulation tasks (Hafner et al., 2019). Recent work has demonstrated the effectiveness of Stable Diffusion models in generating realistic game states to serve as world models (Alonso et al., 2024).

2.2. Video Diffusion Models

With the rapid advancement of diffusion models (Ho et al., 2020), video generation has made significant strides (Ho et al., 2022). The field has evolved from traditional U-Net-based architectures (Ronneberger et al., 2015) to Transformer-based frameworks, DiT, (Peebles & Xie, 2023), enabling video diffusion models to generate highly realistic and temporally coherent videos. Recently, autoregressive video generation (Chen et al., 2024) has emerged as a promising approach to extend video length, theoretically indefinitely. Notably, Diffusion Forcing (Chen et al., 2024) introduces a per-frame noise-level denoising paradigm. Unlike the full-sequence paradigm, which applies a uniform noise level across all frames, per-frame noise-level denoising offers a more flexible approach, enabling autoregressive generation.

3. Method

3.1. Model Architecture

In this work, we evaluate two diffusion models, OASIS and WorldMem, that serve as a world model of Minecraft. They have the same architecture because WorldMem was trained by borrowing the OASIS architecture and fine-tuning its weights. The architecture consists of a VAE, a DiT with an addition on the Spatio-Temporal transformer block for better scalability with respect to the number of frames, and Diffusion Forcing that allows training on past frames with a variant noise level for better auto-regressive inference. The model size is 500M params. The context length of the model is 32 frames.

3.2. Data

In this section, we cover what the two models in question were trained on:

1. OASIS was trained from scratch on the VPT (Baker et al., 2022) dataset that consists of trajectories of real players solving complex tasks. We call this dataset VPT.

2. WorldMem fine-tuned OASIS on a 20M frame dataset collected by the authors using a random agent that walks and spins in various biomes. It's worth noting that the camera action values in this dataset are limited to $\{-15, 0, +15\}$ degrees instead of the full range of $[-180, +180]$ as in the VPT dataset. This dataset represents a much simpler distribution than VPT's. We call this dataset WorldMem.

Since we want to see how each model does not only on the distribution it was trained on, but also on other distributions in the environment, for evaluation, we use three datasets representing different distributions. We draw two of them from the VPT and WorldMem datasets by randomly sampling 128 256-frame segments from the held-out test splits of the VPT and WorldMem datasets described above. The third dataset, called Consistency, we programmatically collect using MineRL (Guss et al., 2019). The purpose of this dataset is to test how the models perform on trajectories that should have an element of consistency within the trajectory and usually are not present in either of the two existing datasets.

The Consistency dataset is comprised of two types of tasks: cycle and reverse. The idea is to make the agent do a set of actions that produce frames, with some of them being equivalent to each other. For example, in the cycle task, with cycle length being n, frames $0, n, 2n, ...$ should be equal, and so are $1, n+1, 2n+1, ...$, and so on. In the reverse task with the length of the forward sequence being n, frames $0, 2n$ should be equal, and so are $1, 2n-1$ and so on. The average number of trajectory frames is 50, and the number of trajectories is 100 (Click for an example video.). This, when we compare the sequence of generated frames to the ground truth frames, allows us to test the model's self-consistency. To match the camera action distribution of the WorldMem dataset, we limit the camera action values to $\{-15, 0, +15\}$ degrees in the Consistency dataset as well.

4. Experiments & Results

We consider two models:

1. OASIS: a model trained on the VPT dataset from scratch.

2. WorldMem: a model trained by fine-tuning OASIS on the WorldMem dataset.

To analyze how each model performs, we evaluate them on three datasets: VPT, WorldMem, and Consistency. It's worth noting that VPT and Consistency datasets follow the same format, and WorldMem has a different format. We implement dedicated dataset classes to support the two formats. The same goes for models. OASIS model was trained with camera actions compressed from the full, $[-180, +180]$, range to $[-1, 1]$ range, whereas WorldMem was trained on camera actions represented as a discrete set of $\{-1, 0, 1\}$, corresponding to $\{-15, 0, +15\}$ degrees. To support both input formats, we implement the corresponding converters where, in the case of WorldMem, we divide the degree value by 15.

During evaluation, we run each model autoregressively with 20 DDIM steps, starting with the first ground truth frame of the trajectory and replaying all of its actions. To provide a quantitative comparison, we calculate a per-frame PSNR between the generated frames and the ground truth.

MODEL	VPT PSNR	WORLDMEM PSNR	CONSISTENCY PSNR
OASIS	18.90	10.32	16.76
WORLDMEM	13.73	15.00	10.44

Table 1. Average PSNR values for OASIS and WorldMem models on VPT, WorldMem, and Consistency datasets.

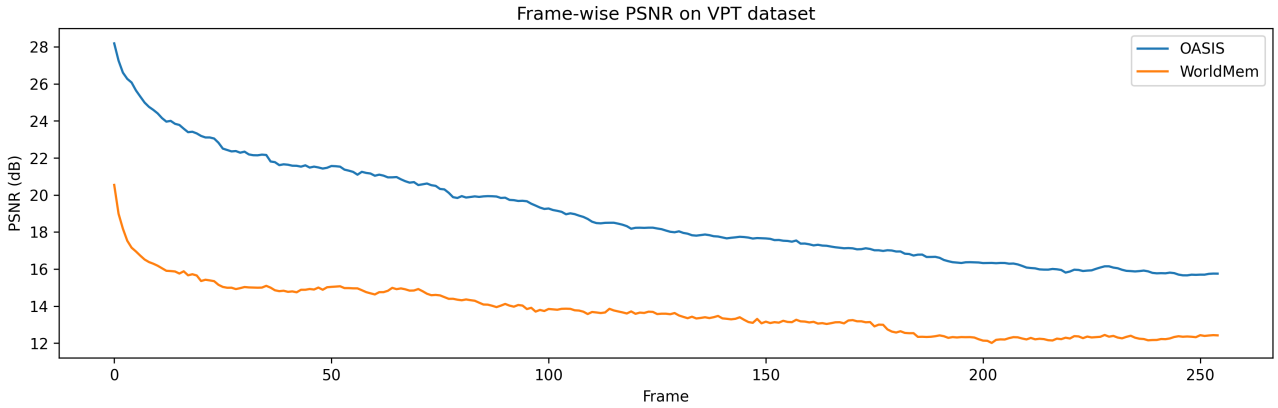

Figure 1. The PSNR-per-step curve on the VPT dataset.

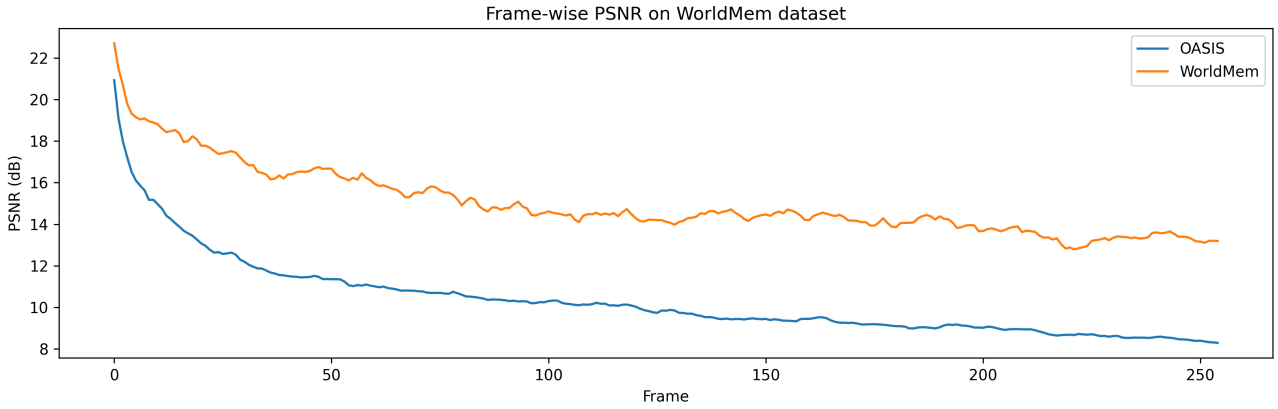

Figure 2. The PSNR-per-step curve on the World dataset.

We provide links to six generated videos (one video for each model and dataset) in Appendix A. The model trained from scratch, OASIS, does reasonably well on the challenging, diverse VPT dataset, but quickly turns into a blur on the WorldMem dataset. The finetuned model, WorldMem, does very well on the WorldMem dataset, but quickly collapses on the challenging VPT dataset. Both models quickly turn into a blur on the Consistency dataset.

From the quantitative analyses in Table 1, Figures 1,2, and 3, and qualitative analyses, we see that the OASIS model does poorly on the simple, limited WorldMem dataset, not being able to generalize to a different distribution of the environment. More interestingly, the WorldMem model, although having been pre-trained on the VPT data and then exposed to the additional WorldMem data, performs badly on the VPT test dataset. We argue this happens due to the catastrophic forgetting effect commonly observed in large language models finetuning.

We draw two conclusions. First, the current video game world models are unable to produce meaningful results on a distribution, even a simple one, different from the one they were trained on. To produce truly generalizable, robust

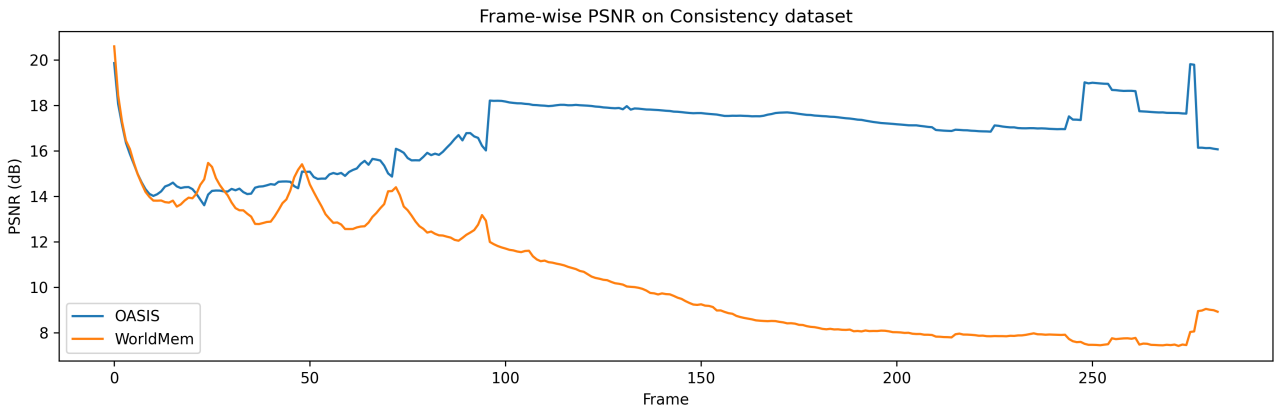

Figure 3. The PSNR-per-step curve on the Consistency dataset.

models there needs to be a fundamental change in either the model size, training dataset size, or the generative paradigm. Second, to extend the capabilities of an existing world model to a new distribution, finetuning on this new distribution alone doesn't work. Instead, one should merge the existing and new distributions into one dataset and fine-tune on it.

5. Conclusion

In this work, we evaluate the performance of two recent world models of Minecraft, OASIS and WorldMem, on three datasets, VPT, WorldMem, and Consistency, representing different environment distributions. We show that the current world models of complex environments with vast action spaces are very sensitive to the data distribution they were trained on and cannot generalize to a different distribution of the same environment. Moreover, we show that finetuning a model trained on one distribution on another distribution leads to catastrophic forgetting and bad performance on the original distribution.

References

Alonso, E., Jelley, A., Micheli, V., Kanervisto, A., Storkey, A., Pearce, T., and Fleuret, F. Diffusion for world modeling: Visual details matter in atari. *arXiv preprint arXiv:2405.12399*, 2024.

Baker, B., Akkaya, I., Zhokhov, P., Huizinga, J., Tang, J., Ecoffet, A., Houghton, B., Sampedro, R., and Clune, J. Video pretraining (vpt): Learning to act by watching unlabeled online videos, 2022. URL https://arxiv.org/abs/2206.11795.

Chen, B., Monso, D. M., Du, Y., Simchowitz, M., Tedrake, R., and Sitzmann, V. Diffusion forcing: Next-token prediction meets full-sequence diffusion, 2024. URL https://arxiv.org/abs/2407.01392.

Decart and Julian Quevedo, Quinn McIntyre, S. C. X. C. R. W. Oasis: A universe in a transformer. 2024. URL https://oasis-model.github.io/.

Guss, W. H., Houghton, B., Topin, N., Wang, P., Codel, C., Veloso, M., and Salakhutdinov, R. Minerl: A large-scale dataset of minecraft demonstrations, 2019. URL https://arxiv.org/abs/1907.13440.

Ha, D. and Schmidhuber, J. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

Hafner, D., Lillicrap, T., Ba, J., and Norouzi, M. Dream to control: Learning behaviors by latent imagination, 2020. URL https://arxiv.org/abs/1912.01603.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models, 2022. URL https://arxiv.org/abs/2204.03458.

Kim, S. W., Zhou, Y., Philion, J., Torralba, A., and Fidler, S. Learning to simulate dynamic environments with gamegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1231–1240, 2020.

Peebles, W. and Xie, S. Scalable diffusion models with transformers, 2023. URL https://arxiv.org/abs/2212.09748.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation, 2015. URL https://arxiv.org/abs/1505.04597.

Wang, G., Xie, Y., Jiang, Y., Mandlekar, A., Xiao, C., Zhu, Y., Fan, L., and Anandkumar, A. Voyager: An open-ended embodied agent with large language models, 2023. URL https://arxiv.org/abs/2305.16291.

Xiao, Z., Lan, Y., Zhou, Y., Ouyang, W., Yang, S., Zeng, Y., and Pan, X. Worldmem: Long-term consistent world simulation with memory, 2025. URL https://arxiv.org/abs/2504.12369.

A. Experiment Details

MODEL	VPT VIDEO	WORLDMEM VIDEO	CONSISTENCY VIDEO
OASIS	CLICK	CLICK	CLICK
WORLDMEM	CLICK	CLICK	CLICK

Table 2. Video sample links for OASIS and WorldMem models generated on VPT, WorldMem, and Consistency datasets.