Poster: #13

Learning to Play: Game Generation Model with Action and Memory

Feiyan Zhou, Zhaoyang Lu, Qingzheng Gao, Peize Sun (PI)

Supervisor: Prof. Amjad Tehmina, Prof Smruthi Mukund

Northeastern University
Khoury College of Computer Sciences

Introduction

Generative game models aim to create interactive virtual environments directly from data, replacing traditional game engines that rely heavily on manual design.

Recent advances in diffusion-based video generation have motivated treating game generation as **next-frame prediction** conditioned on past states and user inputs.

Leading work such as **DeepMind's Genies** and **WorldLabs' Marble** demonstrates **real-time (≈24 fps)** generation with **minute-long** temporal consistency, highlighting the growing potential of AI-driven interactive worlds.

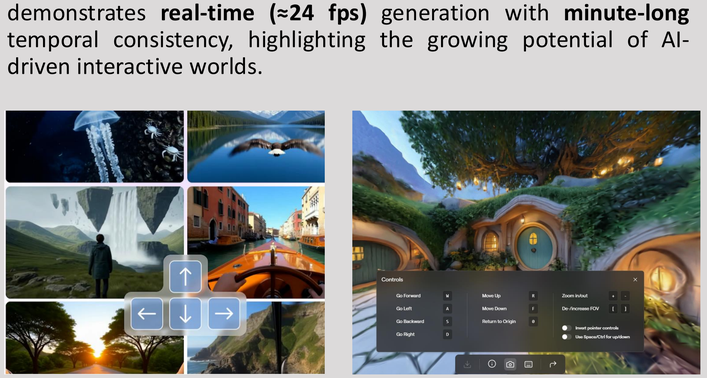


Fig 1. Google DeepMind Genie3 & World Labs 3D World Model Generation

Problem Statement

Despite the progress of diffusion models, generating playable games remains difficult due to three core limitations:

- **Limited Memory:** Models struggle to maintain stable game dynamics across long sequences.
- **Interaction Correctness:** Frames don't accurately reflect player actions.
- **Real-Time Constraints:** Diffusion inference is computationally expensive makes real-time difficult to achieve on typical hardware.

Fig 2. Frame with wrong pattern

Approach

Data collection: We gather Super Mario gameplay trajectories from both **real players** and reinforcement-learning **AI agents**, enabling the model to learn diverse game patterns.

Model Pipeline: we employ a **Diffusion Forcing pipeline**[1] that ensures consistent frame-to-frame rendering. A **Variational Autoencoder (VAE)** encodes each frame into a compact latent representation, while a **Latent Diffusion Model (LDM)** learns to approximate the underlying game transition dynamics, predicting the next observation as: $P(o_{t+1} \mid o_t, a_t)$ [2].

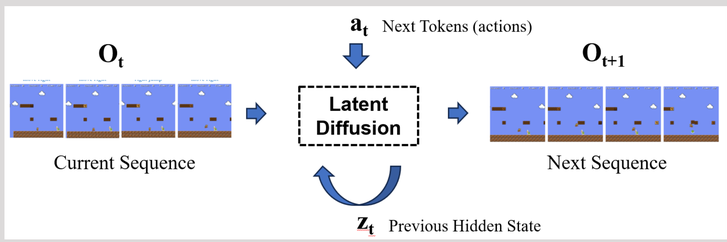


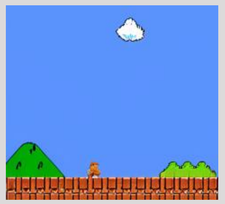
Fig 3. Long Sequence Generation with Diffusion Forcing Framework

Results

Dataset: 400 000+ Super Mario game frames with action information.

VAE Training: We employ a **VAE** with **4× down sampling ratio** as the image encoder–decoder, compressing game frames into a high-quality. The VAE reaches **35.98 dB PSNR**, indicating high-fidelity compression.

Based on our evaluation, the system performs reliably with a **diffusion scaling factor of 0.7064**. While small objects show slight blurring and deformation, the rest of the scenes are well compressed and reconstructed.



General VAE Reconstruction

Reconstruction by Our 4× VAE

Latent Diffusion with Memory: The latent diffusion is augmented with an RNN-style memory that conditions each denoising step on the player action and the previous hidden state.

We train the model using 1× A100 80GB GPUs, with a global **batch size of 120**. After **60 epochs**, the model successfully converges and is able to generate **over 1 minutes** of continuous gameplay video at **16 fps** inference, correctly responding to player actions.

Fig 4. Training Results and Action-Conditioned Inference (Move Right & Jump)

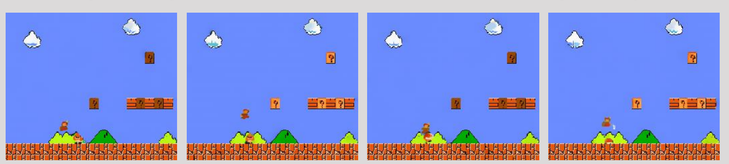


Fig 5. Complex Gameplay Reasoning: Mario Jumping and Stomping the Goomba

Conclusion

Our lightweight diffusion-forcing pipeline runs at **~16 fps on a T4 GPU,** producing coherent, action-conditioned gameplay. It maintains **consistent gameplay video for nearly 1,000 frames**, demonstrating strong potential for efficient and controllable game generation.

Future Work

High-quality, large-scale data are essential for reliable reasoning. While state-of-the-art systems use up to **50M frames**[2]. Improving the dataset through **data augmentation**, or **stronger agent policies**, could further enhance the game transition accuracy.

References

[1] B. Chen, D. M. Monso, Y. Du, M. Simchowitz, R. Tedrake, and V. Sitzmann, "Diffusion Forcing: Next-token Prediction Meets Full-Sequence Diffusion," arXiv preprint arXiv:2407.01392, 2024.
[2]M. Yang, J. Li, Z. Fang, S. Chen, Y. Yu, Q. Fu, W. Yang, and D. Ye, "Playable game generation," arXiv preprint arXiv:2412.00887, 2024.