

Humanoid World Models : Open World Foundation Models for Humanoid Robotics

Qasim Ali ^{*1} Aditya Sridhar ^{*1} Shahbuland Matiana ¹ Alex Wong ¹ Mohammad Al-Sharman ¹

Abstract

Humanoid robots have the potential to perform complex tasks in human-centered environments but require robust predictive models to reason about the outcomes of their actions. We introduce Humanoid World Models (HWM), a family of lightweight, open-source video-based models that forecast future egocentric observations conditioned on actions. We train two types of generative models—Masked Transformers and Flow-Matching—on 100 hours of humanoid demonstrations. Additionally, we explore architectural variants with different attention mechanisms and parameter-sharing strategies. Our parameter-sharing techniques reduce model size by 33–53% with minimal impact on performance or visual fidelity. HWM is designed to be trained and deployed in practical academic and small-lab settings, such as 1–2 GPUs.

1. Introduction

Autonomous humanoid robots have the potential to transform both industry and daily life by automating tasks that are dull, dangerous, or physically demanding (Goswami & Vadakkepat, 2019). Their human-like morphology allows them to operate in spaces built for people, interact naturally with humans, and easily learn from teleoperated demonstrations. Realizing this promise, however, requires more than hardware: these agents must be equipped with sophisticated models of reasoning to navigate the complexity and unpredictability of real-world environments.

While large multimodal models (Bommasani et al., 2022; Zhang et al., 2024; Kim et al., 2024; Black et al., 2024) show some promise in generalization and reasoning, they often lack the accuracy, reliability, and robustness needed for embodied AI in open-world settings (Tong et al., 2024;

Pfeifer & Iida, 2004). As a result, they struggle to meet the demands of embodied agents that must act safely and effectively in dynamic, unstructured environments (Pfeifer & Iida, 2004; Li et al., 2024; Duan et al., 2024).

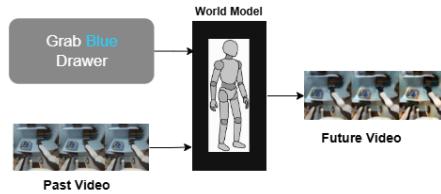


Figure 1. Overview of Humanoid World Model.

One avenue for improving humanoid intelligence and control is through *World Models*—predictive models trained to forecast future outcomes based on past observations and actions (Ha & Schmidhuber, 2018). In our setting, these function as action-conditioned video generators: they predict future visual states as sequences of frames, enabling agents to simulate the consequences of candidate actions (Du et al., 2023). By simulating the outcomes of actions before trying them in the real world, world models support long-horizon planning, counterfactual reasoning, and data-efficient policy learning through synthetic rollouts (Yang et al., 2024; 2023).

Despite recent progress in video generation, most models are built for entertainment applications (Liu et al., 2024; Polyak et al., 2024), emphasizing visual appeal over physical, ego-centric plausibility. Many world models remain closed-source, require large-scale compute for training or inference, or are not designed for humanoid robots (Yang et al., 2023; NVIDIA et al., 2025; Zhu et al., 2024). As a result, there is a clear gap: few open-source models are both physically grounded for humanoid robots and lightweight enough to train or run inference on modest academic hardware (e.g., 2–3 GPUs). To address this, we ask: Can we build a physically plausible, humanoid-specific world model that runs on just two GPUs?

We introduce **Humanoid World Models** (HWM), an open-source lightweight world model for humanoid robotics on 100 hours of humanoid video demonstrations. We inves-

^{*}Equal contribution ¹University of Waterloo, Waterloo, Canada. Correspondence to: Qasim Ali <m45ali@uwaterloo.ca>, Aditya Sridhar <a27sridh@uwaterloo.ca>.

tigate two distinct video generation paradigms: Masked Transformers and Flow-Matching models. Drawing from recent advances in image and video generation, we explore four architectural variations within each framework. These variations are defined along two axes: (1) joint vs. cross-attention mechanisms, and (2) shared vs. separate parameters across token streams. This design space is motivated by successful architecture strategies from the much more explored text-to-image models.

Our experiments show that, given our dataset and compute constraints, Masked Transformers consistently outperformed Flow-Matching models—even when the latter used more parameters and were trained for longer. Across both families of models, we observed that the different architectural variants performed comparably in most scenarios. However, key trends emerged: within the Masked Transformer framework, the joint attention variant achieved the best overall performance. In contrast, for Flow-Matching models, split attention proved most effective.

Importantly, we found that parameter-sharing strategies yielded near-identical performance to their non-shared counterparts while reducing parameter counts by 33–53%, significantly lowering computational requirements. These results suggest that architectural and efficiency trade-offs—particularly attention design and parameter sharing—can be leveraged to build lightweight, performant world models without compromising quality.

Related Works

1.1. Humanoid Robots

Humanoid robots (Humanoids) are actuated, bipedal, and bimanual robots designed to anthropomorphically resemble the human body structure (Goswami & Vadakkepat, 2019). Their morphology is well-suited for operating in human-centered environments, enabling effective interaction and physical compatibility with everyday settings. This embodiment facilitates deployment across a wide range of real-world domains, including households (Imtiaz & Khan, 2024), manufacturing facilities (Hirose & Ogawa, 2007), elderly care homes (Imtiaz & Khan, 2024), and clinical or medical environments (Goswami & Vadakkepat, 2019). Beyond physical compatibility with environments, the human-like form of humanoids allows for more natural interaction with people and the ability to imitate human behaviors (Vianello et al., 2021). Furthermore, collecting demonstration data is relatively straightforward for these platforms, as they can directly mimic human motions and tasks (Zhao et al., 2023). Our world models are trained specifically for humanoids, but can easily be extended to other embodiments.

1.2. World Models

A core challenge in robotics is enabling agents to perceive their environment, reason over possible actions, and plan goal-directed behavior—especially in novel or unstructured settings. Foundation models (Bommasani et al., 2022) attempt to generalize across tasks through large-scale, multimodal training. Recent approaches include Vision-Language-Action (VLA) models that predict actions from video and language inputs (Black et al., 2024; Kim et al., 2024), and large language or vision-language models used as high-level planners (Liang et al., 2023; Li et al., 2024; Wang et al., 2024). However, these methods often struggle with spatial reasoning (Tong et al., 2024), continuous sensorimotor processing, and require complex prompting strategies (Li et al., 2024).

In contrast, video generation models offer a more grounded alternative for embodied agents. Video captures fine-grained physical and temporal structure that language alone cannot express (Yang et al., 2024). When trained to predict future video frames given past observations and actions, these models serve as *World Models*—internal simulators that forecast the outcomes of potential action sequences (Ha & Schmidhuber, 2018; Du et al., 2023). This enables agents to plan through imagination, reason counterfactually, and generate synthetic experience for policy training.

In this work, we develop a humanoid-specific world model implemented as a video generator: it predicts future egocentric video frames from past video and action sequences. This generative modeling approach allows us to simulate plausible futures, supporting both planning and data-efficient learning in complex environments.

1.3. Video Generation Models

Early video generation relied on GANs (Goodfellow et al., 2014), but training instability limited their use. Recent methods improve both quality and efficiency by training generative models like diffusion models (Sohl-Dickstein et al., 2015; Rombach et al., 2022) or masked transformers (Vaswani et al., 2023) in the compressed latent spaces of Variational Autoencoders (VAEs) (Harvey et al., 2022). Discrete latent approaches like vector quantized VAEs (VQ-VAEs) (van den Oord et al., 2018) enable token-based video generation using masked or autoregressive transformers (Ramesh et al., 2021; Chang et al., 2022b). Flow Matching (Lipman et al., 2023) offers an alternative to diffusion with simpler training and faster sampling while preserving sample quality.

Masked Video Generation: Transformer-based approaches have been widely used for image and video generation. These methods operate in the finite and quantized spaces of VQ-VAEs. MaskGIT (Chang et al., 2022b) introduced

a masked, bidirectional transformer for image generation from VQ-VAE latents, along with a non-autoregressive decoding scheme that significantly reduced sampling time compared to diffusion and autoregressive methods. Masked token prediction offers two key advantages over autoregressive approaches (Ramesh et al., 2021; Wu et al., 2024): (1) bidirectional context across space and time improves representation learning, and (2) tokens can be decoded in parallel, greatly accelerating inference. MAGVIT (Yu et al., 2023) extended this approach to video using spatio-temporal VQ-VAEs, while MAGVIT2 (Yu et al., 2024) introduced a stronger tokenizer that outperformed diffusion-based baselines with fewer sampling steps. Open-MAGVIT2 (Luo et al., 2024) provides an open-source implementation. We explore similar non-autoregressive masked video transformers for building humanoid-specific world models.

Diffusion and Flow-Matching: Video diffusion models (Ho et al., 2022) extend diffusion processes to sequences of images, modeling both spatial and temporal dynamics. However, the added temporal dimension significantly increases computational cost. Recent advances in spatiotemporal VAEs (Xing et al., 2024; Bar-Tal et al., 2024) mitigate this by compressing video into low-dimensional latent spaces, making training and inference more tractable.

Large-scale text-to-video diffusion models such as Sora (Liu et al., 2024), CogVideoX (Yang et al., 2025), and MovieGen (Polyak et al., 2024) achieve impressive visual quality but are designed for entertainment and lack support for conditioning on past video—a key requirement for physically grounded, ego-centric prediction.

Several recent works explore video generation for robotic agents, but most are not designed for humanoid platforms and are not open source. UniSim (Yang et al., 2023) trains a text-conditioned video diffusion model for zero-shot policy transfer, but it relies on an outdated U-Net backbone and is not open source. IraSim (Zhu et al., 2024) uses a factorized spatial-temporal transformer within a diffusion framework, but targets robot arms and lacks temporal compression, relying instead on frame-wise image VAEs. Navigation World Models (Bar et al., 2025) are limited to low-DoF mobile robots and generate individual future frames rather than continuous video sequences.

NVIDIA’s Cosmos (NVIDIA et al., 2025) is a notable exception—an open-source, high-fidelity video-to-video model. However, it is not designed for humanoid embodiments and is prohibitively resource-intensive. Its smallest variant (7B parameters) requires 8 NVIDIA H100 GPUs for training and over 40GB of VRAM for inference. On our compute setup of 2 NVIDIA A6000s, generating 121 frames video took over an hour, making finetuning or deployment impractical.

We explore training a lightweight flow-matching model in

the continuous latent space of Cosmos’s VAE (NVIDIA et al., 2025).

1.4. Video Transformer Architectures

For the Masked Video Model, we follow prior work in using factorized spatio-temporal attention (Bruce et al., 2024; Xiang et al., 2024) to reduce computational overhead. This approach separates spatial and temporal attention to scale more efficiently with video length.

In the broader diffusion and flow-matching video generation models literature, architectural trends have shifted from convolutional U-Nets (Ho et al., 2020) to transformer-based designs for improved scalability (Peebles & Xie, 2023). In image generation, joint attention blocks—where image and context tokens are processed together—are now standard, as seen in Stable Diffusion 3 (SD3) (Esser et al., 2024). Subsequent work has shown that complexity can be further reduced through parameter sharing across token streams (fal.ai Blog, 2024; Chen et al., 2025).

In contrast, most video diffusion models still avoid joint attention due to its high memory cost over long video sequences. Instead, they typically use a two-stage attention scheme: self-attention over video tokens, followed by cross-attention with context (Polyak et al., 2024; NVIDIA et al., 2025). Leading models such as Cosmos (NVIDIA et al., 2025) follow this structure, while more complex designs like mixture-of-experts (Kong et al., 2025) or pyramidal transformers (Jin et al., 2024) further increase system complexity.

In our work, we revisit joint attention for video generation. Joint attention over spatiotemporal video tokens is now feasible thanks to two key advances: (1) highly compressive VAEs that reduce spatial resolution (e.g. factor of 16) and temporal resolution (e.g. factor of 8), and (2) efficient parameter-sharing techniques from recent image generation models (fal.ai Blog, 2024; Chen et al., 2025).

2. Methodology

We develop two humanoid-specific world models based on distinct generative paradigms: *Masked Humanoid World Model* (Masked-HWM) and *Flow Humanoid World Model* (Flow-HWM). Masked-HWM employs masked video modeling in a discrete latent space (via VQ-VAE), while Flow-HWM uses flow matching in a continuous latent space. We detail the video generation frameworks and architectures in 2.2 and 2.3 respectively, but describe the transformer block design in detail in 2.4.

2.1. Formulation

The goal is to predict plausible future video frames given a sequence of past video frames and associated actions. Formally, the model predicts a sequence of f future RGB frames $v_f \in \mathbb{R}^{f \times 3 \times H \times W}$, conditioned on p past frames $v_p \in \mathbb{R}^{p \times 3 \times H \times W}$, p past actions $a_p \in \mathbb{R}^z$, and f future actions $a_f \in \mathbb{R}^z$. Action vectors include joint angles, velocities, and gripper states of the humanoid.

Following prior works (Rombach et al., 2022; Chang et al., 2022a), we train our generative models in a VAE’s compressed latent space: v_p and v_f are encoded into latent representations L_p and L_f . Masked-HWM uses VQ-VAE to quantize latents into tokens from a finite vocabulary of size s , while Flow-HWM uses a continuous VAE.

2.2. Masked Video Modelling

We train the Masked-HWM variant using the Masked Video Modelling (MVM) paradigm (Chang et al., 2022a; Yu et al., 2023). After passing v_p, v_f through a VQ-VAE to yield L_p, L_f , we concatenate the past and future latent tokens $L = [L_p; L_f]$ along the temporal dimension.

During training, Masked-HWM receives corrupted and masked versions of the latent sequence as input. Following Copilot-4D (Zhang et al., 2023), we add noise to the latents by corrupting L with random token replacements at a rate uniformly sampled from $\mathcal{U}(0, \rho_{\max})$, where ρ_{\max} denotes the maximum corruption rate. Next, we apply masking to the future latents L_f using a per-frame thresholding strategy. For each frame, we sample a value $r \sim \mathcal{U}(0, 1)$ and compute a masking threshold $\gamma(r)$ using a predefined scheduling function. Then, for each token in the future sequence L_f , we sample a probability from $\mathcal{U}(0, 1)$ and mask the token if it falls below the frame’s threshold.

The model is trained to reconstruct the original tokens at these masked positions. Let M denote the binary mask indicating which tokens have been masked within corrupted L_f . The training objective is to minimize the cross-entropy loss between the predicted tokens \hat{L}_f at the masked locations M and the true tokens at those same locations, as follows:

$$\mathcal{L} = -\mathbb{E}_M \left[\sum_i M_i \log p(\hat{L}_i | L_f) \right]$$

where \hat{L}_i is the model’s prediction and $M_i \in \{0, 1\}$ indicates whether location i was masked. This loss encourages the model to accurately reconstruct corrupted or hidden tokens in the future sequence.

At inference time, we begin by masking all tokens in the future latent sequence L_f . Generation proceeds latent frame by latent frame: after predicting one frame’s tokens, the

result is fed back to help condition the next. Within each frame, tokens are predicted in parallel over K refinement steps. At each step, a random subset of tokens is re-masked and re-predicted, allowing the model to iteratively improve its guesses. This parallel decoding strategy significantly accelerates generation compared to traditional autoregressive methods, which decode tokens sequentially. For more details on the sampling procedure, we refer readers to MaskGiT (Chang et al., 2022a).

2.2.1. ARCHITECTURE

Tokenization The input video frames are encoded using a VQ-VAE performing spatiotemporal compression, yielding discrete tokens. Each latent pixel is treated as a token and projected into h -dimensions. Action sequences (a_p, a_f) are independently embedded into the same h -dimensional space using a multi-layer perceptron (MLP), ensuring compatibility with the video tokens.

Transformer: The resulting four token streams—past video (v_p), future video (v_f), past actions (a_p), and future actions (a_f)—are fed into a stack of d transformer blocks. After processing, the video tokens are linearly projected to a distribution over the VQ-VAE codebook of size s , representing the predicted token identities. Details of the transformer block variants are provided in Section 2.4.

2.3. Flow Matching for Video Generation

We train the Flow-HWM variant using the Flow Matching (FM) framework (Lipman et al., 2023; Albergo et al., 2023; Liu et al., 2022). The framework formulates video generation as a continuous transformation of samples from a simple prior distribution (Gaussian noise) into data samples drawn from the target distribution. As opposed to learning a reversed stochastic process like in traditional diffusion models (Song et al., 2021), FM directly learns a time-dependent velocity field that drives this transformation.

Let \mathbf{X}_1 denote a video sample in the latent space, and let $\mathbf{X}_0 \sim \mathcal{N}(0, \mathbf{I})$ represent a random sample from the Gaussian prior. We train the model by sampling an intermediate time $t \in [0, 1]$ and construct a point along the trajectory \mathbf{X}_t using linear interpolation:

$$\mathbf{X}_t = t\mathbf{X}_1 + (1 - (1 - \sigma_{\min})t)\mathbf{X}_0, \quad (1)$$

where σ_{\min} is a small positive constant ensuring non-zero support at $t = 1$. The ground-truth velocity of the transformation path is then given by the time derivative:

$$\mathbf{V}_t = \frac{d\mathbf{X}_t}{dt} = \mathbf{X}_1 - (1 - \sigma_{\min})\mathbf{X}_0. \quad (2)$$

Our model, parameterized by θ , predicts the instantaneous

velocity field $u_\theta(\mathbf{X}_t, \mathbf{P}, t)$ conditioned on the past video frames v_p , past actions a_p , future actions a_f \mathbf{P} and time t . The training objective of the model is to minimize the expected mean squared error between the predicted and ground-truth velocity:

$$\mathbb{E}_{t, \mathbf{X}_0, \mathbf{X}_1, a_p, a_f, v_p} = \left[\|u_\theta(\mathbf{X}_t, a_p, a_f, v_p, t) - \mathbf{V}_t\|^2 \right]. \quad (3)$$

We adopt classifier-free guidance (Ho & Salimans, 2022) to improve conditional generation by enabling the model to better balance conditioning signals from actions and past context during training and inference. During inference, generation proceeds by integrating the learned velocity field from $t = 0$ to $t = 1$, starting from pure Gaussian noise and employing the first-order Euler ODE solver.

2.3.1. ARCHITECTURE

Tokenization We tokenize the compressed latent video frames L_p and L_f by dividing them into $p_{lw} \times p_{lw}$ spatial and p_t temporal segments per token. Each token is projected to h channels via a convolutional layer. Action sequences a_p and a_f are embedded using an MLP into the same h -dimensional space. The timestep t is encoded using sinusoidal embeddings following DDPM (Ho et al., 2020).

Transformer After tokenization, each of the four streams of tokens (v_f, v_p, a_f, a_p) are kept separate and processed by d transformer blocks sequentially. In the final layer, we apply time modulation as described in DDPM, followed by a linear projection of the future tokens v_f from h dimensions back to l latent dimensions. The resulting tokens are then reshaped into the original video’s spatiotemporal format to be decoded back to pixel space the VAE. We detail the transformer block design in 2.4.

2.4. Transformer Block Design

We evaluate several architectural variants of the transformer block used in both Masked-HWM and Flow-HWM, focusing on three key design dimensions: (1) attention structure (joint vs. split attention), (2) parameter sharing across token streams, and (3) token stream grouping (modality-based vs. fully separate).

Base Block: We start by describing the Base Transformer Block, which is augmented to create other block designs. It processes four token streams—past video (v_p), future video (v_f), past actions (a_p), and future actions (a_f)—each with its own set of parameters but a shared joint attention layer.

The Base Block design for the Masked-HWM is illustrated in Figure 2. Following prior work in non-autoregressive video generation methods (Bruce et al., 2024; Xiang et al., 2024), we adopt a factorized or separate spatial and tempo-

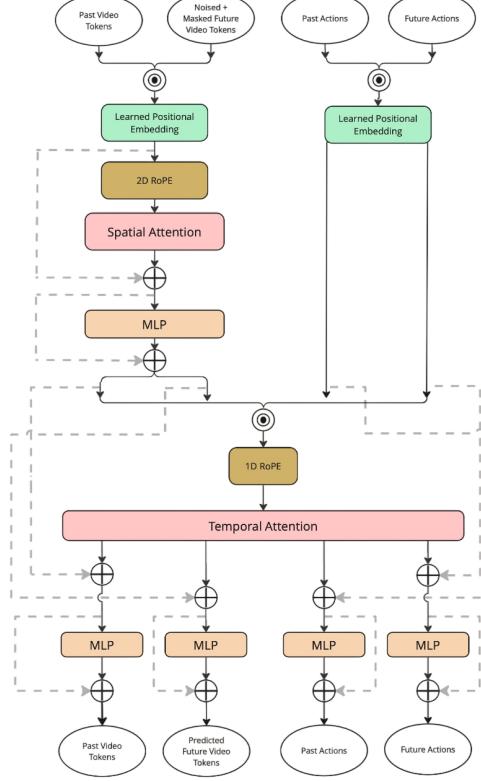


Figure 2. Architecture of a single transformer block in Masked-HWM (Base Block variant). Video and action streams are processed independently, with video streams also receiving Spatial Attention. All streams interact via joint Temporal Attention. RoPE is applied per attention type (2D for spatial and 1D for temporal). Each stream uses distinct MLP weights in the feedforward stage.

ral attention layers. Compared to full spatiotemporal attention, factorized attention reduces the computational cost and scales more efficiently with video length. During temporal attention, all tokens from various streams [a_p, a_f, \mathbf{X}] jointly attend to one another along the temporal dimension. Spatial attention is applied separately only to the video tokens. Rotary Position Embeddings (RoPE)(Su et al., 2023) are used during spatial and temporal attention.

The Base Block for the Flow-HWM variant, as illustrated in Figure 3, is inspired by Stable Diffusion 3 (SD3)(Esser et al., 2024). This design processes the four token streams— v_p , v_f , a_p , and a_f —with separate parameters and enables interaction through a joint attention operation. Within each block, each stream is first modulated by the timestep using learned scale α_0 and shift β_0 parameters(Peebles & Xie, 2023). Subsequently, stream-specific queries, keys, and values are computed using separate W_{QKV} projections. We add positional encodings to the queries and keys of each token stream. Specifically, we add 3D Rotary Position Embeddings (RoPE) to the video tokens, as done in Cosmos

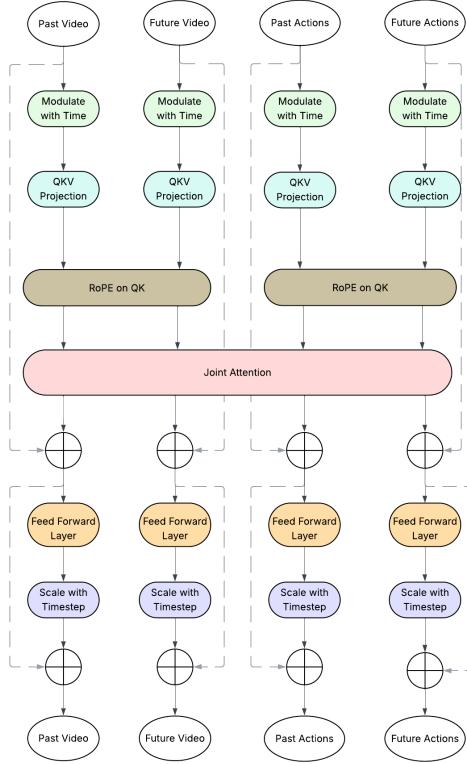


Figure 3. Architecture of a single transformer block in Flow-HWM (Base Block variant). Each token stream (past/future video and actions) uses separate weights for timestep modulation, QKV projection, and feedforward MLPs. Joint Attention integrates all streams. RoPE is applied by modality: 3D for video tokens, 1D for action tokens.

(NVIDIA et al., 2025), and apply 1D RoPE across time for the action tokens. Past and future tokens are concatenated prior to adding positional encoding.

A joint multi-stream attention operation is then applied across all tokens, followed by another timestep-dependent rescaling using γ_0 . In the feedforward stage, tokens are again modulated with the timestep embedding using new parameters α_1 and β_1 , passed through a stream-specific MLP, and finally rescaled using γ_1 . Residual connections are added during the attention and the feedforward stages.

Parameter Sharing: Recent advances in efficient diffusion transformer design (Chen et al., 2025; fal.ai Blog, 2024) have showcased that the benefits of joint attention can be garnered with far fewer parameters using shared attention. We evaluate two parameter-sharing strategies. These strategies selectively share key transformer components across token streams, including timestep modulation parameters (α, β, γ), QKV projection weights (W_{QKV}), and feedforward MLPs.

In the *Full Sharing* variant, the modulation scalars, QKV projections, and MLPs are shared across all four token streams (v_p, v_f, a_p, a_f). This maximally reduces parameter count and compute overhead. In the *Modality Sharing* variant, parameters are shared within each modality, i.e., video streams (v_p, v_f) share weights, and action streams (a_p, a_f) share weights. This strikes a balance between model compactness and representational flexibility.

In our implementation, we retain the original *Base Block* configuration (with fully separate parameters) for the first four layers and apply parameter sharing in the remaining $l - 4$ transformer layers, following practices from (fal.ai Blog, 2024). This hybrid scheme allows early layers to learn modality-specific representations, while deeper layers focus on cross-modal reasoning in a more compact parameter regime.

Split Attention. While joint attention across all token streams enables rich cross-modal interactions, it becomes increasingly expensive with longer video sequences and higher token counts. To address this, we implement a two-stage *Split Attention* mechanism, which has been widely adopted in recent large-scale video generation models (NVIDIA et al., 2025; Bar et al., 2025; Polyak et al., 2024) for its computational efficiency.

In this variant, each stream—future video (v_f), past video (v_p), past actions (a_p), and future actions (a_f)—first undergoes *independent self-attention* within its own sequence. This allows each modality and temporal context to process intra-stream dependencies with minimal overhead.

Following self-attention, we apply a *cross-attention* operation in which the future video tokens v_f serve as queries, and the keys and values are drawn from the remaining streams (v_p, a_p, a_f). This structure enables the model to selectively condition future video generation on past observations and intended actions, while avoiding the full cost of global attention. As in the joint attention variant, both self- and cross-attention layers are modulated using time-dependent scaling and shifting, with parameters (α, β, γ) learned per stream.

3. Results

Dataset We train our models on the 1xGPT dataset (1X Technologies, 2024), which contains 100 hours of egocentric video captured from the Humanoid EVE Android executing various tasks. Video frames are recorded at 30 Hz, with each frame paired with a corresponding action vector $a \in \mathbb{R}^{25}$ representing movement velocities, hand closure states, and pitch-yaw-roll angles of the joints (wrist, knee, elbow, shoulder, neck, and hip). We train both models to generate $f = 8$ future frames conditioned on $p = 9$ past frames at $H = 256 \times W = 256$ resolution.

Table 1. Performance of Masked-HWM variants. Base Block yields the best FID; Split Attention gives the highest PSNR. Parameter sharing improves efficiency with minimal quality loss.

METRIC	SPLIT ATTENTION	BASE BLOCK	MODALITY SHARING	FULL SHARING
MODEL SIZE (BILLION)	0.220	0.321	0.237	0.195
PEAK GPU MEMORY (GB)	2.22	2.63	2.30	2.12
SAMPLES PER SECOND	2.09	2.27	2.25	2.36
FID	15.31	10.13	11.67	14.21
PSNR (dB)	29.37	29.02	28.97	28.66

Evaluation We evaluate our models using Fréchet Inception Distance (FID) (Heusel et al., 2018) and PSNR (Horé & Ziou, 2010), computed over 21,000 generated frames from a held-out validation set. To isolate the impact of the generative model from the VAE’s reconstruction quality, all ground truth frames are passed through the same VAE encoder-decoder used during training. In addition to image quality, we report model parameter count, video latent decoding speed (measured in samples per second), and peak GPU memory usage during inference.

3.1. Masked Video Modelling

Experimental Setup We tokenize all video frames using the NVIDIA Cosmos DV8x8x8 tokenizer, which applies $8\times$ compression in both spatial and temporal dimensions, reducing 256×256 RGB frames to 32×32 latent grids. Each training sample consists of 2 fully unmasked past latents and 1 partially masked future latent. We use a cosine masking schedule from MaskGIT and inject Copilot-4D style noise with a uniform corruption rate sampled from $\mathcal{U}(0, \rho_{\max})$, where $\rho_{\max} = 0.2$.

Models are trained for 60,000 steps using the AdamW optimizer, with a learning rate linearly decayed from $3e-5$ after 100 warmup steps. We use 24 transformer layers, 8 heads, 512-dimensional tokens, and an MLP hidden size of 2048. We apply standard normal initialization ($\mu = 0, \sigma = 0.02$) for all weights, except Xavier initialization for the mask token and output projection. Training is performed on a single NVIDIA A6000 with batch size 16. During inference, we use $K = 2$ decoding iterations.

3.1.1. QUALITATIVE RESULTS

Sample videos from the Base Block variant are shown in Figure 4. The model learns both structural elements of the scene, such as furniture and small objects that the robot is manipulating, and overall textures. Larger parts of the robot’s appendage, including arms and wheels, are generally modeled accurately. However, precision elements, such as fingers, are often slightly blurry and entangled. Further, the visual quality of generated images is robust to lighting, as shown in the 3rd (lighter) and 4th (darker) sequences.



Figure 4. Four sample videos from the Base Block Variant of Masked-HWM. Top row: generated frames; bottom row: ground truth.

3.1.2. QUANTITATIVE RESULTS

Table 1 reports the performance of different Masked-HWM variants. The *Base Block* achieved the best FID score (10.13), indicating the strongest visual quality among all configurations. The *Split Attention* variant obtained the highest PSNR (29.37 dB), but slightly underperformed in FID, suggesting better pixel-level fidelity at the cost of less realistic global structure.

Both parameter-sharing variants—*Modality Sharing* and *Full Sharing*—reduced model size and memory while maintaining competitive FID and PSNR. *Modality Sharing* matched Base Block quality with 26% fewer parameters, showing that intra-modality sharing suffices. *Full Sharing*, though slightly lower in quality, had the smallest footprint and fastest inference (2.36 samples/sec), making it ideal for efficiency-focused settings.

3.2. Flow Matching

Experimental Setup.

We use the Cosmos Continuous 8x16x16 tokenizer that spa-

Table 2. Performance of Flow-HWM variants. Full Sharing achieves the best FID, memory usage, and speed even when compared to the Base Variant; Split Attention yields the highest PSNR.

METRIC	SPLIT ATTENTION	BASE BLOCK	MODALITY SHARING	FULL SHARING
MODEL SIZE (BILLION)	0.944	1.36	0.886	0.648
PEAK GPU MEMORY (GB)	4.37	5.94	4.41	3.25
SAMPLES PER SECOND	1.11	1.69	1.89	1.91
FID	111.12	111.59	112.75	110.73
PSNR (dB)	20.50	20.42	20.50	20.43

tially compresses frames by a factor of 16 (from 256×256 to 16×16), and performs $8 \times$ temporal compression. Using a more spatially compressive VAE relative to Masked-HWM allows for joint attention while using much larger models required for flow-matching networks. Models are trained with $d = 17$ transformer layers and $h = 1172$ -dimensional tokens using the AdamW optimizer, a learning rate of $1e-4$, cosine learning rate scheduler, and batch size 128. Training runs for 150,000 steps across 2 NVIDIA A6000 GPUs. We use patch sizes $p_{lw} = 2, p_t = 1$. We initialize final linear layers with Xavier initialization, as zero-initialization (as in prior works (Peebles & Xie, 2023)) led to instability. No learning rate warmup is used, as it degraded convergence. During inference, we apply 50 denoising steps with a classifier-free guidance scale of 3.0.

3.2.1. QUALITATIVE RESULTS

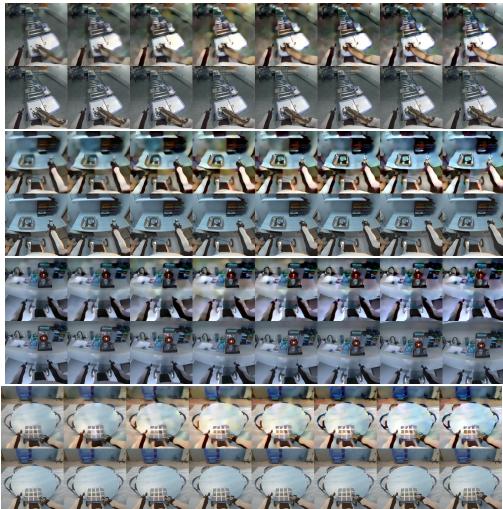


Figure 5. Sample videos from the Base Variant of Flow-HWM. Top row: generated frames; bottom row: ground truth.

As shown in Figure 5, the generated videos successfully capture overall scene structure (walls, floors, and doors). However, the outputs exhibit noticeable blurriness and artifacting like spotted patches. Visual quality tends to degrade in later frames, with the model struggling to preserve straight edges and rounded shapes. While the arms are often rendered with

high fidelity, the model defaults to a canonical arm appearance and fails to represent unusual or out-of-distribution arm configurations (as seen in the top video strip).

3.2.2. QUANTITATIVE RESULTS

Table 2 summarizes Flow-HWM performance. *Full Sharing* offered the best trade-off, outperforming the Base Block in all metrics while halving the parameter count. It achieved the lowest FID (110.73), fastest inference (1.91 samples/sec), and minimal memory use (3.25 GB). *Modality Sharing* delivered similar quality with notable efficiency gains, while *Split Attention* yielded the highest PSNR (20.50 dB) but required more memory and was slower. Overall, parameter sharing improves efficiency without compromising quality.

Overall, none of the Flow-HWM variants outperformed the Masked-HWM models in either visual quality or sampling speed, suggesting that masked video modeling is a more effective generative paradigm for our dataset and compute constraints. The results consistently show that parameter sharing—particularly in the Full Sharing variant—provides substantial efficiency gains with minimal loss in visual fidelity, very beneficial in resource-constrained settings.

4. Conclusion

Humanoid World Models demonstrate that it is possible to build physically plausible, efficient predictive models tailored for humanoid robotics using modest computational resources. Through effective parameter-sharing strategies and compressive VAEs, HWM enables open-world reasoning for embodied agents in compute-constrained settings.

Impact Statement

This work advances video-based world models for humanoid robots, aiming to make predictive simulation more computationally accessible. While our models support progress in embodied AI, they are not intended for unsafe or unethical deployment.

References

- 1X Technologies. 1X World Model Challenge, June 2024.
- Albergo, M. S., Boffi, N. M., and Vanden-Eijnden, E. Stochastic interpolants: A unifying framework for flows and diffusions, 2023. URL <https://arxiv.org/abs/2303.08797>.
- Bar, A., Zhou, G., Tran, D., Darrell, T., and LeCun, Y. Navigation world models, 2025. URL <https://arxiv.org/abs/2412.03572>.
- Bar-Tal, O., Chefer, H., Tov, O., Herrmann, C., Paiss, R., Zada, S., Ephrat, A., Hur, J., Liu, G., Raj, A., Li, Y., Rubinstein, M., Michaeli, T., Wang, O., Sun, D., Dekel, T., and Mosseli, I. Lumiere: A space-time diffusion model for video generation, 2024. URL <https://arxiv.org/abs/2401.12945>.
- Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L. X., Tanner, J., Vuong, Q., Walling, A., Wang, H., and Zhilinsky, U. pi0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. URL <https://arxiv.org/abs/2410.24164>.
- Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosse-lut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the opportunities and risks of foundation models, 2022. URL <https://arxiv.org/abs/2108.07258>.
- Bruce, J., Dennis, M. D., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.
- Chang, H., Zhang, H., Jiang, L., Liu, C., and Freeman, W. T. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11315–11325, 2022a.
- Chang, H., Zhang, H., Jiang, L., Liu, C., and Freeman, W. T. Maskgit: Masked generative image transformer, 2022b. URL <https://arxiv.org/abs/2202.04200>.
- Chen, C., Qian, R., Hu, W., Fu, T.-J., Tong, J., Wang, X., Li, L., Zhang, B., Schwing, A., Liu, W., and Yang, Y. Dit-air: Revisiting the efficiency of diffusion model architecture design in text to image generation, 2025. URL <https://arxiv.org/abs/2503.10618>.
- Du, Y., Yang, M., Florence, P., Xia, F., Wahid, A., Ichter, B., Sermanet, P., Yu, T., Abbeel, P., Tenenbaum, J. B., Kaelbling, L., Zeng, A., and Tompson, J. Video language planning, 2023. URL <https://arxiv.org/abs/2310.10625>.
- Duan, J., Pumacay, W., Kumar, N., Wang, Y. R., Tian, S., Yuan, W., Krishna, R., Fox, D., Mandlekar, A., and Guo, Y. Aha: A vision-language-model for detecting and reasoning over failures in robotic manipulation, 2024. URL <https://arxiv.org/abs/2410.00371>.
- Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Lacey, K., Goodwin, A., Marek, Y., and Rombach, R. Scaling rectified flow transformers for high-resolution image synthesis, 2024. URL <https://arxiv.org/abs/2403.03206>.
- fal.ai Blog. Auraflow: Generate high-fidelity 3d assets with diffusion models. <https://blog.fal.ai/auraflow/>, Apr 2024. Accessed April 19, 2025.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks, 2014. URL <https://arxiv.org/abs/1406.2661>.
- Goswami, A. and Vadakkepat, P. (eds.). *Humanoid Robotics: A Reference*. Springer Dordrecht, 2019. ISBN 978-94-007-6046-2. doi: 10.1007/978-94-007-6046-2. URL <https://link.springer.com/referencework/10.1007/978-94-007-6046-2>.

- Ha, D. and Schmidhuber, J. World models. *CoRR*, abs/1803.10122, 2018. URL <http://dblp.uni-trier.de/db/journals/corr/corr1803.html#abs-1803-10122>.
- Harvey, W., Naderiparizi, S., and Wood, F. Conditional image generation by conditioning variational auto-encoders, 2022. URL <https://arxiv.org/abs/2102.12037>.
- Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. URL <https://arxiv.org/abs/1706.08500>.
- Hirose, M. and Ogawa, K. Honda humanoid robots development. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1850):11–19, 2007. doi: 10.1098/rsta.2006.1917. URL <https://royalsocietypublishing.org/doi/10.1098/rsta.2006.1917>.
- Ho, J. and Salimans, T. Classifier-free diffusion guidance, 2022. URL <https://arxiv.org/abs/2207.12598>.
- 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>.
- Horé, A. and Ziou, D. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pp. 2366–2369, 2010. doi: 10.1109/ICPR.2010.579.
- Imtiaz, R. and Khan, A. Perceptions of humanoid robots in caregiving: A study of skilled nursing home and long term care administrators, 2024. URL <https://arxiv.org/abs/2401.02105>.
- Jin, Y., Sun, Z., Li, N., Xu, K., Xu, K., Jiang, H., Zhuang, N., Huang, Q., Song, Y., Mu, Y., and Lin, Z. Pyramidal flow matching for efficient video generative modeling, 2024. URL <https://arxiv.org/abs/2410.05954>.
- Kim, M., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., Vuong, Q., Kollar, T., Burchfiel, B., Tedrake, R., Sadigh, D., Levine, S., Liang, P., and Finn, C. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.
- Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., Wu, K., Lin, Q., Yuan, J., Long, Y., Wang, A., Wang, A., Li, C., Huang, D., Yang, F., Tan, H., Wang, H., Song, J., Bai, J., Wu, J., Xue, J., Wang, J., Wang, K., Liu, M., Li, P., Li, S., Wang, W., Yu, W., Deng, X., Li, Y., Chen, Y., Cui, Y., Peng, Y., Yu, Z., He, Z., Xu, Z., Zhou, Z., Xu, Z., Tao, Y., Lu, Q., Liu, S., Zhou, D., Wang, H., Yang, Y., Wang, D., Liu, Y., Jiang, J., and Zhong, C. Hunyuandvideo: A systematic framework for large video generative models, 2025. URL <https://arxiv.org/abs/2412.03603>.
- Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.
- Li, W., Yu, Z., She, Q., Yu, Z., Lan, Y., Zhu, C., Hu, R., and Xu, K. Llm-enhanced scene graph learning for household rearrangement, 2024. URL <https://arxiv.org/abs/2408.12093>.
- Liang, J., Huang, W., Xia, F., Xu, P., Hausman, K., Ichter, B., Florence, P., and Zeng, A. Code as policies: Language model programs for embodied control, 2023. URL <https://arxiv.org/abs/2209.07753>.
- Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling, 2023. URL <https://arxiv.org/abs/2210.02747>.
- Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL <https://arxiv.org/abs/2209.03003>.
- Liu, Y., Zhang, K., Li, Y., Yan, Z., Gao, C., Chen, R., Yuan, Z., Huang, Y., Sun, H., Gao, J., He, L., and Sun, L. Sora: A review on background, technology, limitations, and opportunities of large vision models, 2024. URL <https://arxiv.org/abs/2402.17177>.
- Luo, Z., Shi, F., Ge, Y., Yang, Y., Wang, L., and Shan, Y. Open-magvit2: An open-source project toward democratizing auto-regressive visual generation, 2024. URL <https://arxiv.org/abs/2409.04410>.
- NVIDIA, :, Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., Dworakowski, D., Fan, J., Fenzi, M., Ferroni, F., Fidler, S., Fox, D., Ge, S., Ge, Y., Gu, J., Gururani, S., He, E., Huang, J., Huffman, J., Jannaty, P., Jin, J., Kim, S. W., Klár, G., Lam, G., Lan, S., Leal-Taixe, L., Li, A., Li, Z., Lin, C.-H., Lin, T.-Y., Ling, H., Liu, M.-Y., Liu, X., Luo, A., Ma, Q., Mao, H., Mo, K., Mousavian, A., Nah, S., Niverty, S., Page, D., Paschalidou, D., Patel, Z., Pavao, L., Ramezanali, M., Reda, F., Ren, X., Sabavat, V. R. N., Schmerling, E., Shi, S., Stefaniak, B., Tang, S., Tchapmi, L., Tredak, P., Tseng, W.-C., Varghese, J., Wang, H., Wang, H., Wang, H., Wang,

- T.-C., Wei, F., Wei, X., Wu, J. Z., Xu, J., Yang, W., Yen-Chen, L., Zeng, X., Zeng, Y., Zhang, J., Zhang, Q., Zhang, Y., Zhao, Q., and Zolkowski, A. Cosmos world foundation model platform for physical ai, 2025. URL <https://arxiv.org/abs/2501.03575>.
- Peebles, W. and Xie, S. Scalable diffusion models with transformers, 2023. URL <https://arxiv.org/abs/2212.09748>.
- Pfeifer, R. and Iida, F. *Embodied Artificial Intelligence: Trends and Challenges*, pp. 1–26. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-27833-7. doi: 10.1007/978-3-540-27833-7_1. URL https://doi.org/10.1007/978-3-540-27833-7_1.
- Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.-Y., Chuang, C.-Y., Yan, D., Choudhary, D., Wang, D., Sethi, G., Pang, G., Ma, H., Misra, I., Hou, J., Wang, J., Jagadeesh, K., Li, K., Zhang, L., Singh, M., Williamson, M., Le, M., Yu, M., Singh, M. K., Zhang, P., Vajda, P., Duval, Q., Girdhar, R., Sumbaly, R., Rambhatla, S. S., Tsai, S., Azadi, S., Datta, S., Chen, S., Bell, S., Ramaswamy, S., Sheynin, S., Bhattacharya, S., Motwani, S., Xu, T., Li, T., Hou, T., Hsu, W.-N., Yin, X., Dai, X., Taigman, Y., Luo, Y., Liu, Y.-C., Wu, Y.-C., Zhao, Y., Kirstain, Y., He, Z., He, Z., Pumarola, A., Thabet, A., Sanakoyeu, A., Mallya, A., Guo, B., Araya, B., Kerr, B., Wood, C., Liu, C., Peng, C., Vengertsev, D., Schonfeld, E., Blanchard, E., Juefei-Xu, F., Nord, F., Liang, J., Hoffman, J., Kohler, J., Fire, K., Sivakumar, K., Chen, L., Yu, L., Gao, L., Georgopoulos, M., Moritz, R., Sampson, S. K., Li, S., Parmeggiani, S., Fine, S., Fowler, T., Petrovic, V., and Du, Y. Movie gen: A cast of media foundation models, 2024. URL <https://arxiv.org/abs/2410.13720>.
- Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation, 2021. URL <https://arxiv.org/abs/2102.12092>.
- Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models, 2022. URL <https://arxiv.org/abs/2112.10752>.
- Sohl-Dickstein, J., Weiss, E. A., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics, 2015. URL <https://arxiv.org/abs/1503.03585>.
- Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations, 2021. URL <https://arxiv.org/abs/2011.13456>.
- Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding, 2023. URL <https://arxiv.org/abs/2104.09864>.
- Tong, S., Liu, Z., Zhai, Y., Ma, Y., LeCun, Y., and Xie, S. Eyes wide shut? exploring the visual shortcomings of multimodal llms, 2024. URL <https://arxiv.org/abs/2401.06209>.
- van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning, 2018. URL <https://arxiv.org/abs/1711.00937>.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need, 2023. URL <https://arxiv.org/abs/1706.03762>.
- Vianello, L., Penco, L., Gomes, W., You, Y., Anzalone, S. M., Maurice, P., Thomas, V., and Ivaldi, S. Human-humanoid interaction and cooperation: a review. *Current Robotics Reports*, 2(4):441–454, December 2021. doi: 10.1007/s43154-021-00068-z. URL <https://doi.org/10.1007/s43154-021-00068-z>.
- Wang, S., Han, M., Jiao, Z., Zhang, Z., Wu, Y. N., Zhu, S.-C., and Liu, H. Llm3:large language model-based task and motion planning with motion failure reasoning, 2024. URL <https://arxiv.org/abs/2403.11552>.
- Wu, J., Yin, S., Feng, N., He, X., Li, D., Hao, J., and Long, M. ivideogpt: Interactive videogpts are scalable world models. In *Advances in Neural Information Processing Systems*, 2024.
- Xiang, J., Liu, G., Gu, Y., Gao, Q., Ning, Y., Zha, Y., Feng, Z., Tao, T., Hao, S., Shi, Y., Liu, Z., Xing, E. P., and Hu, Z. Pandora: Towards general world model with natural language actions and video states, 2024. URL <https://arxiv.org/abs/2406.09455>.
- Xing, Z., Feng, Q., Chen, H., Dai, Q., Hu, H., Xu, H., Wu, Z., and Jiang, Y.-G. A survey on video diffusion models, 2024. URL <https://arxiv.org/abs/2310.10647>.
- Yang, M., Du, Y., Ghasemipour, K., Tompson, J., Schuurmans, D., and Abbeel, P. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 2023.
- Yang, S., Walker, J., Parker-Holder, J., Du, Y., Bruce, J., Barreto, A., Abbeel, P., and Schuurmans, D. Video as the new language for real-world decision making, 2024. URL <https://arxiv.org/abs/2402.17139>.
- Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., Yin, D.,

Zhang, Y., Wang, W., Cheng, Y., Xu, B., Gu, X., Dong, Y., and Tang, J. Cogvideox: Text-to-video diffusion models with an expert transformer, 2025. URL <https://arxiv.org/abs/2408.06072>.

Yu, L., Cheng, Y., Sohn, K., Lezama, J., Zhang, H., Chang, H., Hauptmann, A. G., Yang, M.-H., Hao, Y., Essa, I., and Jiang, L. Magvit: Masked generative video transformer, 2023. URL <https://arxiv.org/abs/2212.05199>.

Yu, L., Lezama, J., Gundavarapu, N. B., Versari, L., Sohn, K., Minnen, D., Cheng, Y., Birodkar, V., Gupta, A., Gu, X., Hauptmann, A. G., Gong, B., Yang, M.-H., Essa, I., Ross, D. A., and Jiang, L. Language model beats diffusion – tokenizer is key to visual generation, 2024. URL <https://arxiv.org/abs/2310.05737>.

Zhang, J., Huang, J., Jin, S., and Lu, S. Vision-language models for vision tasks: A survey, 2024. URL <https://arxiv.org/abs/2304.00685>.

Zhang, L., Xiong, Y., Yang, Z., Casas, S., Hu, R., and Urtasun, R. Copilot4d: Learning unsupervised world models for autonomous driving via discrete diffusion. *arXiv preprint arXiv:2311.01017*, 2023.

Zhao, T. Z., Kumar, V., Levine, S., and Finn, C. Learning fine-grained bimanual manipulation with low-cost hardware, 2023. URL <https://arxiv.org/abs/2304.13705>.

Zhu, F., Wu, H., Guo, S., Liu, Y., Cheang, C., and Kong, T. Irasim: Learning interactive real-robot action simulators. *arXiv:2406.12802*, 2024.