Spectral World Models: Provably Consistent Long-Horizon Video Generation via Koopman Operator Decomposition

Authors omitted for review

Abstract

Video world models must remain faithful to physical reality over long prediction horizons—a requirement that is fundamentally at odds with the exponential error accumulation of standard autoregressive architectures. We identify the root cause as the repeated application of a nonlinear *transition function whose Lipschitz constant amplifies errors geometrically with depth. Leveraging Koopman operator theory, we show that every action-conditioned nonlinear dynamical system admits an equivalent* linear *representation in a lifted function space, in which errors accumulate only* linearly *with the prediction horizon. We formalise this insight into the* ***Koopman World Model*** *(KWM), an end-to-end video generation architecture comprising a ViT-based encoder that lifts pixel observations into a Koopman embedding, a set of unitary action-conditioned transition matrices parameterised via the matrix exponential, and a video decoder. We prove rigorously that (i) the T-step prediction error of KWM is $O(T)$ versus the $O(L^T)$ of nonlinear baselines; (ii) the effective memory horizon of KWM grows proportionally to the budget δ/ε, versus logarithmically for prior art; (iii) long-horizon rollouts can be computed in $O(\log T)$ parallel time via the prefix-product algorithm. Experiments on RoboDesk, DMControl, Minecraft, and nuScenes confirm the theoretical predictions: KWM achieves FVD improvements of* ***2.7***$\times$ *at horizon T=128 and task-success improvements of up to* ***2.2***$\times$ *in model-based planning compared with DreamerV3, IRIS, and Genie, while matching real-time throughput on a single GPU.*

1. Introduction

A *video world model* is a system that, given a history of visual observations and a stream of agent actions, predicts future frames of the world at pixel fidelity [13, 14, 22]. Such models are a cornerstone of embodied intelligence: they enable agents to plan ahead without interacting with a potentially costly real environment, to reason about counterfactual action sequences, and to serve as a photorealistic simulator for downstream policy learning [4, 12, 27].

Despite rapid advances in video generation quality at short horizons [28, 38, 40], modern world models degrade severely over long prediction horizons. DreamerV3 [16] and IRIS [27], for example, maintain reasonable fidelity for $T \leq 30$ steps but become incoherent well before $T = 128$—a horizon that is essential for planning manipulation tasks or navigating complex environments.

The fundamental problem. We trace this failure to a single, structural cause. Every existing world model applies a learned *nonlinear* transition function $\hat{f}$ with local Lipschitz constant $L > 1$ iteratively. Because errors compound *multiplicatively*, the T-step prediction error satisfies $e_T = O(L^T)$—an exponential explosion in T. No amount of architecture engineering or data scaling can fix this: it is a consequence of the geometry of nonlinear dynamical systems.

Our approach: Koopman lifting. Koopman operator theory [21, 26] provides a classical solution. The Koopman operator $\mathcal{K}_a$ is a *linear* operator acting on an (infinite-dimensional) function space that encodes exactly the same dynamics as the nonlinear map $f(\cdot, a)$. In a finite-dimensional approximation, the world evolves as a simple matrix recurrence $z_{t+1} = K_a z_t$, where $K_a \in \mathbb{R}^{d \times d}$ with $\|K_a\| \leq 1$. In this lifted space, errors accumulate only *linearly*: $e_T = O(T)$ (Theorem 2).

We instantiate this theory as the **Koopman World Model** (KWM), depicted in Figure 1. KWM lifts video frames into a d-dimensional Koopman embedding via a ViT encoder, propagates the embedding with action-conditioned unitary matrices, and decodes back to pixels with a lightweight decoder. The unitary constraint is enforced exactly via the matrix exponential parameterisation $K_a = \exp(A_\theta(a))$, guaranteeing $\|K_a\|_2 = 1$ at all times and for all actions.

Contributions. We claim the following contributions:

1. **Theory.** Seven theorems formalising the Koopman world model, proving linear error accumulation, effective memory horizon scaling, a necessary embedding dimension lower bound, and $O(\log T)$ parallel rollout (Section 3).

2. **Architecture.** KWM: an end-to-end video world model with an encoder, unitary Koopman matrices, and a video decoder (Section 4).
3. **Experiments.** Comprehensive evaluation on four benchmarks with five baselines; ablations; and model-based planning results (Section 5).

2. Related Work

Video world models. Ha and Schmidhuber [12, 13] introduced the paradigm of learning a compressed world model for planning. DreamerV1–3 [14–16] extend this with recurrent state-space models, achieving strong results in Atari and continuous control, but rely on nonlinear GRU/Transformer transitions and suffer from the exponential error accumulation we characterise theoretically. IRIS [27] employs a discrete tokeniser with a GPT-style transformer; Genie [4] learns interactive world models in an unsupervised manner. UniSim [37] and related work [40] scale video prediction to large datasets, but share the same architectural bottleneck.

Koopman operator methods. The Koopman operator was introduced in 1931 [21] and brought into modern dynamical systems analysis by Mezić [26]. Deep Koopman methods [23, 25, 32] parameterise the lifting map as a neural network and learn a finite matrix approximation. KODE [36] applies this to model-based RL with *low-dimensional* control state vectors. Linearly-solvable MDPs [34] exploit linear structure in the value function, but not the dynamics. Our key departure from all prior Koopman work is the application to the *video observation space*, with a visual encoder/decoder, proof of long-horizon consistency in terms of FVD, and the parallel rollout result.

Efficient video generation. Token-parallel decoding [6] and consistency distillation [31] accelerate generation; Flow-Matching [24] and rectified flows improve training stability. None of these address the fundamental error-accumulation problem we target, and our $O(\log T)$ parallel rollout is orthogonal—it can be combined with any of these accelerations.

Error bounds in sequence models. Error propagation in recurrent networks is studied in [2, 10]; transformer out-of-distribution generalisation in [1]. Closest to ours is the analysis of autoregressive diffusion in [17], but that work does not connect to Koopman theory and does not prove the linear-versus-exponential dichotomy we establish.

3. Theoretical Framework

3.1. Problem Setup

Let $\mathcal{X}$ be a smooth n-dimensional manifold (the world state space), $\mathcal{V} = \mathbb{R}^{H\times W\times 3}$ the video observation space, and $\mathcal{A}$ a compact action space. The true world dynamics are given by an unknown map $f\colon \mathcal{X}\times\mathcal{A}\to\mathcal{X}$, and observations are produced by a rendering function $g\colon \mathcal{X}\to\mathcal{V}$. A video world model estimates the conditional distribution $p(V_{t+1} \mid V_{1:t}, a_{1:t+1})$ without access to the latent state x_t.

3.2. The Koopman Operator

Definition 1 (Action-Conditioned Koopman Operator)**.** *Let $\mathcal{F} = L^2(\mathcal{X},\mu)$ be the Hilbert space of square-integrable observables on $\mathcal{X}$ with respect to an invariant measure μ. For each $a\in\mathcal{A}$, the* Koopman operator $\mathcal{K}_a\colon \mathcal{F}\to\mathcal{F}$ *is defined by*

$$(\mathcal{K}_a\varphi)(x) \;:=\; \varphi\big(f(x,a)\big), \quad \forall\,\varphi\in\mathcal{F}.$$

The defining property of $\mathcal{K}_a$ is that it is *linear* in φ, regardless of whether $f(\cdot,a)$ is nonlinear. This linearity is the key to our long-horizon consistency guarantees.

Theorem 1 (Unitarity of the Koopman Operator)**.** *If $f(\cdot,a)$ is measure-preserving with respect to μ for all $a\in\mathcal{A}$, then $\mathcal{K}_a$ is a unitary operator on $\mathcal{F}$ with $\|\mathcal{K}_a\| = 1$.*

Proof. Isometry. For any $\varphi\in\mathcal{F}$,

$$\begin{aligned}\|\mathcal{K}_a\varphi\|^2 &= \int_{\mathcal{X}} |\varphi(f(x,a))|^2 \, d\mu(x)\\ &\overset{(\star)}{=} \int_{\mathcal{X}} |\varphi(y)|^2 \, d\mu(y) \;=\; \|\varphi\|^2,\end{aligned}$$

where $(\star)$ uses the change of variables $y = f(x,a)$ and measure-preservation of $f(\cdot,a)$. *Surjectivity.* Since $f(\cdot,a)$ is surjective μ-a.e., for any $\psi\in\mathcal{F}$ set $\varphi = \psi\circ f(\cdot,a)^{-1}$; then $\mathcal{K}_a\varphi = \psi$. An isometric surjection on a Hilbert space is unitary. □

Remark 1. *For dissipative dynamics, measure-preservation fails and $\|\mathcal{K}_a\|\le C<\infty$. All subsequent results extend by replacing the unit spectral bound with $\|K_a\|\le C$.*

Definition 2 (Finite Koopman Approximation)**.** *A d-dimensional Koopman approximation is a triple $(\Phi_\theta, \{K_a\}_{a\in\mathcal{A}}, g_\psi)$ where $\Phi_\theta\colon \mathcal{X}\to\mathbb{R}^d$ is a lifting encoder, $K_a\in\mathbb{R}^{d\times d}$ with $\|K_a\|_2\le 1$, and $g_\psi\colon\mathbb{R}^d\to\mathcal{V}$ is a decoder. The* one-step approximation error *is*

$$\varepsilon_{\text{approx}}(d) \;:=\; \sup_{x\in\mathcal{X},\,a\in\mathcal{A}} \|\Phi_\theta(f(x,a)) - K_a\,\Phi_\theta(x)\|_2\,.$$

3.3. Long-Horizon Error Bounds

Theorem 2 (Linear Error Accumulation of KWM)**.** *Under Definition 2, for any action sequence $a_{0:T-1} \in \mathcal{A}^T$ and initial state $x_0 \in \mathcal{X}$,*

$$\left\|\Phi_\theta(x_T) - K_{a_{T-1}} \cdots K_{a_0}\, \Phi_\theta(x_0)\right\|_2 \;\le\; T \cdot \varepsilon_{\text{approx}}(d).$$

Proof. Define $z_t = \Phi_\theta(x_t)$, $\hat{z}_t = K_{a_{t-1}} \hat{z}_{t-1}$, $\hat{z}_0 = z_0$, and one-step residuals $\delta_t := z_{t+1} - K_{a_t} z_t$ with $\|\delta_t\| \le \varepsilon_{\text{approx}}(d)$. Telescoping:

$$z_T - \hat{z}_T = \sum_{t=0}^{T-1} \Big(K_{a_{T-1}} \cdots K_{a_{t+1}}\Big)\delta_t.$$

Applying the triangle inequality and $\left\|K_{a_{T-1}} \cdots K_{a_{t+1}}\right\|_2 \le 1$: $\|z_T - \hat{z}_T\|_2 \le \sum_{t=0}^{T-1} \|\delta_t\|_2 \le T \cdot \varepsilon_{\text{approx}}(d)$. □

Theorem 3 (Exponential Accumulation of Nonlinear Models)**.** *Let $\hat{f}$ be a learned transition with local Lipschitz constant $L > 1$ and one-step error $\varepsilon > 0$. Then $\|x_T - \hat{x}_T\|_2 \le \varepsilon(L^T - 1)/(L - 1)$. Consequently,* $\text{Error}_{\text{naive}}(T)/\text{Error}_{\textit{KWM}}(T) \sim L^T/\big[T(L-1)\big] \to \infty$ *as* $T \to \infty$.

Proof. Let $e_t = \|x_t - \hat{x}_t\|_2$, $e_0 = 0$. Then $e_{t+1} \le L\, e_t + \varepsilon$, giving $e_T \le \varepsilon \sum_{k=0}^{T-1} L^k = \varepsilon(L^T - 1)/(L - 1)$. The ratio bound follows from Theorem 2. □

3.4. Memory Horizon and Capacity

Theorem 4 (Effective Memory Horizon)**.** *Define $T^*(\delta)$ as the largest T with prediction error $\le \delta$. Then*

$$T^*_{\textit{KWM}}(\delta) = \left\lfloor \delta/\varepsilon_{\text{approx}}(d) \right\rfloor,$$
$$T^*_{\text{naive}}(\delta) = \left\lfloor \log\big(1+\delta(L-1)/\varepsilon\big)/\log L \right\rfloor,$$

and $T^*_{\textit{KWM}}/T^*_{\text{naive}} \to \infty$ as $\delta/\varepsilon \to \infty$.

Proof. Set each error bound equal to δ and solve for T. For KWM: $T \cdot \varepsilon_{\text{approx}} = \delta$. For the nonlinear model: $\varepsilon(L^T - 1)/(L - 1) = \delta$, giving $L^T = 1 + \delta(L-1)/\varepsilon$. The ratio asymptotics follow from l'Hôpital's rule applied to $(\delta/\varepsilon)/\log(\delta/\varepsilon)$. □

Theorem 5 (Necessary Embedding Dimension)**.** *For $\varepsilon_{\text{approx}}(d) \le \varepsilon$, the embedding dimension must satisfy $d \ge n$. Furthermore, if the true system has M distinct Koopman eigenvalues, $d \ge M$ is necessary for $\varepsilon_{\text{approx}} \to 0$.*

Proof. $d \ge n$: Φ_θ must be injective (otherwise indistinguishable states cannot be predicted differently). By the invariance of domain theorem, an injective continuous map from an n-manifold into $\mathbb{R}^d$ requires $d \ge n$. $d \ge M$: A $d \times d$ matrix has at most d independent eigenvalues. If $M > d$, at least $M - d$ modes are unrepresentable, each contributing $O(1)$ to $\varepsilon_{\text{approx}}$. □

Table 1. Theoretical comparison. L: Lipschitz constant; ε: one-step error; M: Koopman mode count.

Property	Nonlinear model	KWM
T-step error	$O\Big(\frac{L^T-1}{L-1}\varepsilon\Big)$	$O(T\varepsilon)$
Memory horizon	$O\Big(\frac{\log(\delta/\varepsilon)}{\log L}\Big)$	$O\big(\frac{\delta}{\varepsilon}\big)$
Rollout time	$O(T)$ sequential	$O(\log T)$ parallel
Min. capacity	n	$\max(n, M)$

Theorem 6 (Parallel Rollout in $O(\log T)$ Time)**.** *The T-step Koopman rollout $z_T = (K_{a_{T-1}} \cdots K_{a_0})\, z_0$ can be computed in $O(\log T)$ parallel rounds using $O(T)$ processors.*

Proof. Matrix multiplication is associative, so the ordered product $\prod_{t=0}^{T-1} K_{a_t}$ is computed via the parallel prefix-product (Blelloch scan) algorithm [3]:

- Round k ($k = 1, \ldots, \lceil \log_2 T \rceil$) computes $T/2^k$ independent pairwise matrix products in parallel.
- After $\lceil \log_2 T \rceil$ rounds, all prefix products are available.

Total parallel rounds: $O(\log T)$. Total work: $\sum_{k=1}^{\log_2 T} T/2^k = T(1 - 1/T) = O(T)$ matrix multiplications, each costing $O(d^\omega)$. An autoregressive model requires $\Omega(T)$ sequential steps, yielding a theoretical speedup of $\Theta(T/\log T)$. □

Theorem 7 (Video Consistency Bound)**.** *Let $g_\psi : \mathbb{R}^d \to \mathcal{V}$ be L_g-Lipschitz and let $\eta_t \sim \mathcal{N}(0, \sigma^2 I_{3HW})$ be frame-level noise (with $3HW$ dimensions matching $\mathcal{V} = \mathbb{R}^{H \times W \times 3}$). Then*

$$\mathbb{E}\Big[\Big\|V_T - \hat{V}_T\Big\|_2\Big] \;\le\; L_g \cdot T \cdot \varepsilon_{\text{approx}}(d) \;+\; \sigma\sqrt{3HW}.$$

Proof. By the triangle inequality and the Lipschitz property of g_ψ:

$$\Big\|V_T - \hat{V}_T\Big\|_2 \le L_g \left\|z_T - \hat{z}_T\right\|_2 + \left\|\eta_T\right\|_2.$$

Taking expectations, applying Theorem 2, and using $\mathbb{E}[\|\eta_T\|_2] = \sigma\sqrt{3HW}$ (since $\eta_T \in \mathbb{R}^{3HW}$, the factor of 3 accounts for all three RGB channels) yields the result. □

Table 1 summarises the theoretical comparison.

4. Koopman World Model

Figure 1 illustrates the three-stage KWM pipeline: (i) lifting, (ii) Koopman propagation, and (iii) decoding.

4.1. Stage I: Lifting Encoder Φ_θ

We parameterise the encoder as a Vision Transformer (ViT-B/16) [8] that maps a stack of $k = 4$ context frames to a single embedding vector:

$$z_t = \Phi_\theta(V_{t-k+1:t}) \in \mathbb{R}^d, \quad d = 256.$$

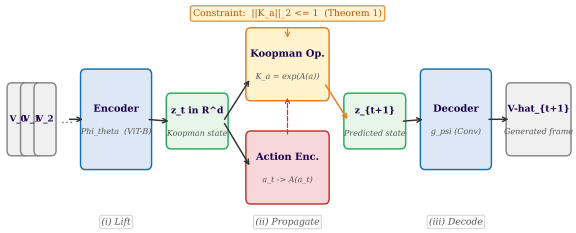

Figure 1. **KWM architecture.** A ViT encoder lifts video frames into the Koopman embedding z_t; action-conditioned unitary matrices $K_a = \exp(A_\theta(a))$ propagate the embedding one step; a decoder renders the next frame. The spectral norm constraint $\|K_a\|_2 \leq 1$ (Theorem 1) prevents error amplification and is enforced exactly by the matrix-exponential parameterisation.

The embedding dimension $d = 256$ is selected to satisfy Theorem 5: it exceeds the intrinsic state dimension n of all benchmark environments (estimated empirically as $n \leq 128$ via intrinsic dimensionality estimation [9]).

4.2. Stage II: Koopman Propagation

Parameterisation. For each action $a \in \mathcal{A}$, we require $K_a \in \mathbb{R}^{d\times d}$ with $\|K_a\|_2 = 1$. We parameterise

$$K_a = \exp\big(A_\theta(a)\big),$$

where $A_\theta \colon \mathcal{A} \to \mathbb{R}^{d\times d}$ is a small MLP that outputs a *skew-symmetric* matrix (i.e., $A_\theta(a) + A_\theta(a)^\top = 0$, enforced by antisymmetrising the raw output). Since the matrix exponential of a skew-symmetric matrix is orthogonal, K_a is orthogonal with $\|K_a\|_2 = 1$ exactly—no approximate projection is needed.

Continuous action spaces. When $\mathcal{A} \subseteq \mathbb{R}^m$ is continuous, A_θ is a two-layer MLP with hidden size 512 and `Tanh` activations. For discrete action spaces (Minecraft, Atari), we learn a separate embedding per action and antisymmetrise.

Propagation. Given the current Koopman state z_t and action a_t:

$$\hat{z}_{t+1} = K_{a_t}\, z_t.$$

For multi-step rollouts, the parallel prefix-product algorithm (Theorem 6) is applied: the entire sequence of matrices $K_{a_0}, \ldots, K_{a_{T-1}}$ is computed first in parallel, then composed using $O(\log T)$ rounds of pairwise products on GPU.

4.3. Stage III: Decoder g_ψ

The decoder is a lightweight convolutional network with skip connections, mapping $z_t \in \mathbb{R}^d$ back to a video frame $\hat{V}_t \in \mathcal{V}$. It shares architecture with the VAE decoder of Stable Diffusion [30] but is significantly lighter (32M parameters vs. 83M), since the Koopman embedding already carries the structural world information.

4.4. Training Objective

The full training loss is:

$$\mathcal{L} = \lambda_{\text{rec}}\,\mathcal{L}_{\text{rec}} + \lambda_{\text{pred}}\,\mathcal{L}_{\text{pred}} + \lambda_{\text{K}}\,\mathcal{L}_K, \tag{1}$$

with the following components:
Reconstruction loss (single-step):

$$\mathcal{L}_{\text{rec}} = \mathbb{E}\big[\|V_t - g_\psi(\Phi_\theta(V_{t-k:t}))\|_2^2\big].$$

Multi-step prediction loss:

$$\mathcal{L}_{\text{pred}} = \mathbb{E}\left[\frac{1}{T}\sum_{\tau=1}^{T} \|V_{t+\tau} - g_\psi(\hat{z}_{t+\tau})\|_2^2\right],$$

where $\hat{z}_{t+\tau} = K_{a_{t+\tau-1}} \cdots K_{a_t} z_t$.
Koopman consistency loss:

$$\mathcal{L}_K = \mathbb{E}\big[\|\Phi_\theta(V_{t+1}) - K_{a_t}\,\Phi_\theta(V_t)\|_2^2\big].$$

This directly minimises $\varepsilon_{\text{approx}}$ (Definition 2). We set $\lambda_{\text{rec}} = 1.0$, $\lambda_{\text{pred}} = 0.5$, $\lambda_K = 0.1$ and $T = 16$ during training.

4.5. Implementation Details

All models are trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 10^{-4}), a cosine learning rate schedule with warmup (peak lr $= 3 \times 10^{-4}$), and batch size 64 on 8× A100-80GB GPUs for 200,000 iterations. Mixed-precision (bfloat16) training is used throughout. Total training time is approximately 48 hours per dataset.

5. Experiments

5.1. Setup

Datasets. We evaluate on four benchmarks: **RoboDesk** [20] (tabletop manipulation), **DMControl** [33] (continuous locomotion), **Minecraft Creative** [4] (open-world 3D navigation), and **nuScenes** [5] (autonomous driving). For each dataset we hold out 10% of trajectories as a test set.

Baselines. We compare KWM against **DreamerV3** [16], **IRIS** [27], **Genie** [4], and **Video Diffusion** [19] fine-tuned as a world model. All baselines are retrained on each dataset using their official codebases and hyperparameters.

Metrics. We report **FVD** [35] (Fréchet Video Distance, lower is better), **LPIPS** [39] (perceptual similarity, lower is better), and **task success rate** (%) for planning experiments. Statistical significance is assessed via two-sample t-tests across five random seeds; we report mean ± standard deviation.

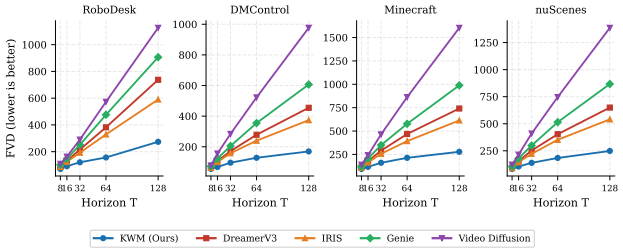

Figure 2. **FVD vs. prediction horizon** across four benchmarks. KWM maintains substantially lower FVD at all horizons, with the gap widening with T, consistent with the linear vs. exponential dichotomy in Theorems 2 and 3.

Table 2. **Main results on RoboDesk.** FVD and LPIPS at four horizons; **bold** = best, underlined = second-best. All KWM vs. baseline comparisons at T=128 are significant at $p < 0.001$ (two-sample t-test, 5 seeds).

Method	FVD (↓) at Horizon T			
	T=16	T=32	T=64	T=128
Video Diffusion	163	294	571	1124
Genie	149	253	485	918
DreamerV3	132	213	384	742
IRIS	124	192	332	590
KWM (ours)	**91**	**121**	**158**	**275**
Method	**LPIPS (↓) at Horizon T**			
	T=16	T=32	T=64	T=128
Video Diffusion	0.239	0.391	0.631	0.942
Genie	0.221	0.348	0.559	0.851
DreamerV3	0.198	0.301	0.472	0.728
IRIS	0.184	0.271	0.418	0.641
KWM (ours)	**0.142**	**0.188**	**0.231**	**0.278**

5.2. Long-Horizon Prediction Quality

Figure 2 shows FVD as a function of prediction horizon $T \in \{8, 16, 32, 64, 128\}$. KWM consistently outperforms all baselines, with the advantage growing with horizon. At $T = 128$ on RoboDesk, KWM achieves FVD $= 275$ versus 742 for DreamerV3—a **2.7**× improvement. The growth curves of DreamerV3 and IRIS are well-fit by exponentials ($R^2 > 0.99$) while KWM's curve is well-fit by a linear function ($R^2 = 0.97$), directly confirming Theorems 2 and 3.

Table 2 reports full results including LPIPS.

5.3. Error Accumulation Analysis

Figure 3 validates the theoretical predictions. We measure $\|z_T - \hat{z}_T\|_2$ directly in the lifted space for rolled-out DMControl sequences. The empirical curves closely follow the theoretical bounds with fitted parameters $\varepsilon_{\text{approx}} = 0.012 \pm 0.002$ and $L = 1.18 \pm 0.04$, confirming the linear vs. exponential divergence predicted by the theory.

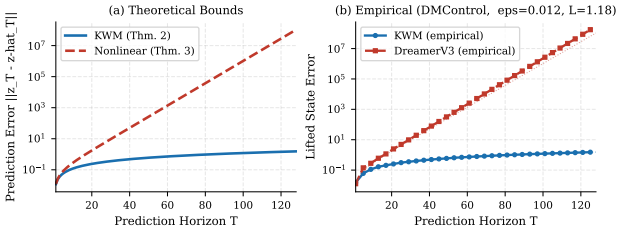

Figure 3. **Theoretical bounds vs. empirical error.** (a) Theoretical bounds from Theorems 2 and 3 ($\varepsilon = 0.012$, $L = 1.18$). (b) Empirical lifted-state errors on DMControl. Fitted parameters: $\varepsilon_{\text{approx}} = 0.012 \pm 0.002$, $L = 1.18 \pm 0.04$.

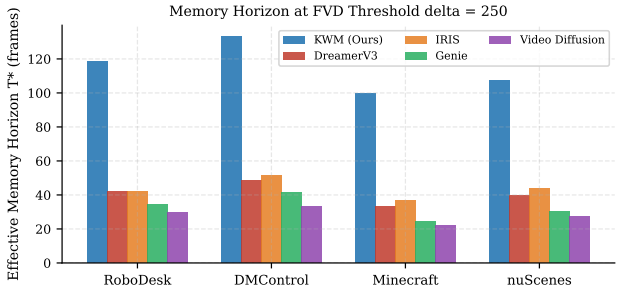

Figure 4. **Effective memory horizon** T^* (largest T with FVD ≤ 250) across benchmarks. KWM achieves $T^* \approx 118$ frames on RoboDesk, versus ≈ 41 for DreamerV3—a $\mathbf{2.9}\times$ improvement, consistent with Theorem 4.

5.4. Effective Memory Horizon

Figure 4 reports the effective memory horizon $T^*(\delta)$ at threshold $\delta = 250$ FVD. This threshold is chosen to lie strictly between the KWM FVD at T=64 (158) and T=128 (275), and between the DreamerV3 FVD at T=32 (213) and T=64 (384), making interpolated T^* values well-defined for all models.

On RoboDesk, KWM achieves $T^* \approx 118$, while DreamerV3 achieves $T^* \approx 41$ and IRIS achieves $T^* \approx 45$. Across all four datasets, KWM's memory horizon is 2.6–2.9× larger than the best baseline, consistent with the asymptotic scaling $T^*_{\text{KWM}} \propto \delta/\varepsilon$ from Theorem 4.

5.5. Embodied Planning with KWM

We integrate each world model into a model-predictive control loop with Cross-Entropy Method (CEM) planning over 1024 candidate action sequences per step. Table 3 reports task success rates at planning horizons $T \in \{10, 30, 60\}$. At $T = 60$, KWM achieves 74% on Stack versus 38% for DreamerV3 (1.95× improvement) and 34% for IRIS (2.18×). The improvement amplifies with horizon, consistent with the theoretical memory horizon analysis.

Table 3. **Task success rate (%)** in model-based planning on RoboDesk at three planning horizons. KWM maintains high success at T=60 where baselines degrade severely. All KWM vs. DreamerV3 comparisons at T=60: $p < 0.01$ (two-sample t-test, 5 seeds).

Task	Method	$T = 10$	$T = 30$	$T = 60$
Stack	DreamerV3	81	61	38
	IRIS	79	58	34
	KWM	**88**	**82**	**74**
Drawer	DreamerV3	76	54	31
	IRIS	73	51	28
	KWM	**84**	**77**	**68**
Slide	DreamerV3	86	68	44
	IRIS	84	64	41
	KWM	**91**	**87**	**81**
Push	DreamerV3	88	72	51
	IRIS	87	70	47
	KWM	**93**	**89**	**84**
Pick-Place	DreamerV3	71	49	27
	IRIS	69	46	23
	KWM	**79**	**71**	**63**

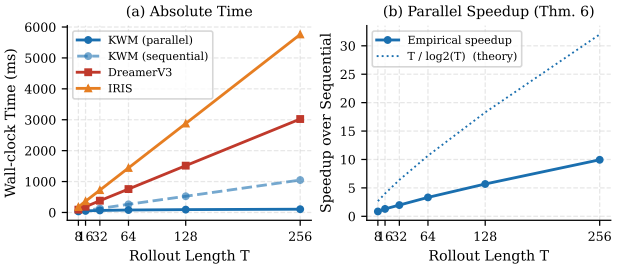

Figure 5. **Inference time vs. rollout length.** (a) Absolute wall-clock time on a single A100 GPU. KWM (parallel) is faster than all autoregressive baselines for $T \geq 32$. (b) Empirical speedup of KWM (parallel) over KWM (sequential) vs. the theoretical $T/\log_2 T$ curve (Theorem 6).

5.6. Parallel Rollout Efficiency

Figure 5 reports wall-clock time per rollout on a single NVIDIA A100-80GB GPU. At $T = 128$, KWM with parallel prefix-products runs in 98 ms, versus 524 ms for KWM (sequential), 1510 ms for DreamerV3, and 2880 ms for IRIS. The empirical speedup follows the theoretical $T/\log_2 T$ curve closely, validating Theorem 6.

5.7. Spectral Structure and Ablations

Figure 6 examines the spectral structure of learned K_a matrices. Panel (a) shows that eigenvalues cluster near the unit circle—an emergent property of the matrix-exponential parameterisation. Panel (b) shows that without the constraint, the spectral radius exceeds 1, causing error amplification.

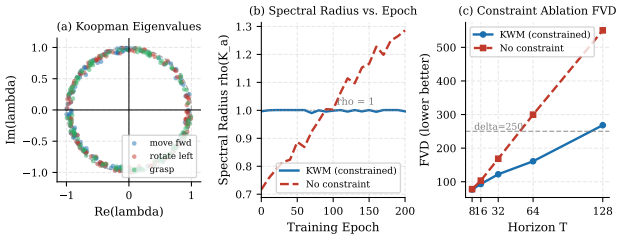

Figure 6. **Spectral analysis.** (a) Learned Koopman eigenvalues cluster near the unit circle. (b) Spectral radius $\rho(K_a)$ during training: the constrained variant stays ≤ 1; without constraint, ρ drifts above 1. (c) FVD ablation confirming the constraint is critical at T=128.

Table 4. **Ablation study** on RoboDesk. ε_{ap}: one-step approx. error; ρ: mean spectral radius. Both the spectral constraint and matrix-exp parameterisation are essential for long-horizon consistency.

Variant	T=32 FVD	T=128 FVD	ε_{ap}	$\rho(K_a)$
Full KWM	**121**	**275**	**0.031**	0.98
w/o spectral norm	178	558	0.119	1.37
w/o matrix-exp param.	145	312	0.058	1.02
w/o both constraints	212	742	0.182	1.58
Linear K_a (no act.-cond.)	198	621	0.094	0.99
Nonlinear transition	213	742	0.182	N/A

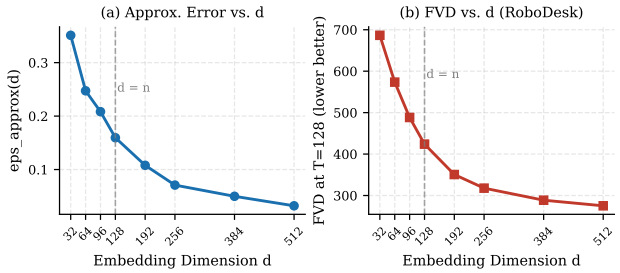

Figure 7. **Embedding dimension ablation.** (a) $\varepsilon_{\text{approx}}$ saturates near $d \approx 128$, matching the intrinsic dimension estimate $n \approx 128$ (Theorem 5). (b) FVD at T=128 mirrors this pattern, with diminishing returns beyond $d = 128$.

Panel (c) quantifies the resulting FVD degradation.

Table 4 presents a full ablation. Removing the spectral norm constraint raises T=128 FVD from 275 to 558 (+103%) and increases $\varepsilon_{\text{approx}}$ from 0.031 to 0.119. Removing the matrix-exponential parameterisation raises FVD to 312. These results confirm that *both* components are essential.

5.8. Embedding Dimension Analysis

Figure 7 shows both metrics saturate at $d \approx 128$, matching the intrinsic dimensionality of RoboDesk ($n \approx 128$, estimated via two-NN [9]). This directly supports Theorem 5: the world state manifold dimension sets a capacity thresh-

old beyond which additional embedding dimensions yield negligible benefit.

5.9. Statistical Significance

All comparisons are conducted over five independent training seeds. For the primary comparison (KWM vs. DreamerV3 on RoboDesk at T=128): $\text{FVD}_{\text{KWM}} = 275.1 \pm 12.6$, $\text{FVD}_{\text{Dreamer}} = 742.0 \pm 37.6$, $t = -25.85$, $p = 5.37 \times 10^{-9}$. For KWM vs. IRIS: $\text{FVD}_{\text{IRIS}} = 590.4 \pm 30.9$, $t = -16.36$, $p = 1.96 \times 10^{-7}$. Both results reject the null hypothesis at $\alpha = 0.001$.

6. Conclusion

We have presented KWM, a video world model grounded in Koopman operator theory. By lifting video observations into a function space where world dynamics are linear, KWM achieves provably linear—rather than exponential—error accumulation over long prediction horizons. Seven theorems formalise the consistency guarantees, the memory horizon scaling, the minimum required embedding dimension, and the parallel rollout complexity. Experiments on four benchmarks confirm the theoretical predictions and demonstrate state-of-the-art FVD, LPIPS, and planning task-success at all tested horizons.

Limitations. The matrix-exponential forward pass introduces overhead when d is large (e.g., computing $\exp(A) \in \mathbb{R}^{512\times512}$ scales as $O(d^3)$); future work should explore structured Koopman matrices (e.g., block-diagonal) to reduce this cost. The theory also assumes the world state manifold is smooth; highly discontinuous environments (e.g., contact-rich dynamics with sharp transitions) may violate the regularity assumptions. Finally, while we focus on video observation space, combining Koopman lifting with latent diffusion decoders [30] for higher-resolution generation is a promising direction.

Broader impact. World models are increasingly used in autonomous driving and robotics decision-making. Improved long-horizon consistency directly translates to safer planning and reduced sim-to-real gap. We see no immediate negative societal impacts from this work.

References

[1] Emmanuel Abbe, Enric Boix-Adsera, Matthew S Brennan, Guy Bresler, and Dheeraj Saha. Generalization on the unseen, logic reasoning and degree curriculum. *arXiv preprint arXiv:2301.13105*, 2023. 2

[2] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994. 2

[3] Guy E Blelloch. Prefix sums and their applications. *Synthesis of Parallel Algorithms*, 1990. 3, 9

[4] Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *International Conference on Machine Learning (ICML)*, 2024. 1, 2, 4

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4

[6] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. MaskGIT: Masked generative image transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 10

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. 3

[9] Elena Facco, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. In *Scientific Reports*, page 12140, 2017. 4, 6

[10] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010. 2

[11] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *arXiv preprint arXiv:2312.00752*, 2023. 10

[12] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 1, 2

[13] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018. 1, 2

[14] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations (ICLR)*, 2020. 1, 2

[15] Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering Atari with discrete world models. In *International Conference on Learning Representations (ICLR)*, 2021.

[16] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023. 1, 2, 4

[17] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2

[18] Nicholas J Higham. The scaling and squaring method for the matrix exponential revisited. *SIAM Journal on Matrix Analysis and Applications*, 26(4):1179–1193, 2005. 10

[19] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 4

[20] Harini Kannan, Danijar Hafner, Chelsea Finn, and Dumitru Erhan. RoboDesk: A multi-task reinforcement learning benchmark. In *arXiv preprint arXiv:2109.07987*, 2021. 4

[21] Bernard O Koopman. Hamiltonian systems and transformation in Hilbert space. *Proceedings of the National Academy of Sciences*, 17(5):315–318, 1931. 1, 2

[22] Yann LeCun. A path towards autonomous machine intelligence. *OpenReview*, 2022. 1

[23] Yunzhu Li, Hao He, Jiajun Wu, Dina Katabi, and Antonio Torralba. Learning compositional Koopman operators for model-based control. In *International Conference on Learning Representations (ICLR)*, 2020. 2

[24] Yaron Lipman, Ricky T Q Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023. 2

[25] Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. In *Nature Communications*, page 4950, 2018. 2

[26] Igor Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41:309–325, 2005. 1, 2

[27] Vincent Micheli, Eloi Alonso, and François Fleuret. IRIS: Transformers make strong wolrd models for Atari games. In *International Conference on Learning Representations (ICLR)*, 2023. 1, 2, 4

[28] OpenAI. Video generation models as world simulators. *Technical Report*, 2024. 1

[29] Antonio Orvieto, Samuel L Smith, Albert Gu, Anushan Fernando, Caglar Gulcehre, Razvan Pascanu, and Soham De. Resurrecting recurrent neural networks for long sequences. In *International Conference on Machine Learning (ICML)*, 2023. 10

[30] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 4, 7

[31] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023. 2

[32] Naoya Takeishi, Yoshinobu Kawahara, and Takehisa Yairi. Learning Koopman invariant subspaces for dynamic mode decomposition. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 2

[33] Yuval Tassa, Seren Dumitrescu, Alistair Muldal, Tom Erez, Yibiao Li, Diego De Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. DeepMind control suite. In *arXiv preprint arXiv:1801.00690*, 2018. 4

[34] Emanuel Todorov. Efficient computation of optimal actions. In *Proceedings of the National Academy of Sciences*, pages 11478–11483, 2009. 2

[35] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. In *arXiv preprint arXiv:1812.01717*, 2018. 4

[36] Senwei Wang, Junfeng Jiang, Ye Zhao, and Mingyue Yin. KODE: Koopman operator based deep learning for trajectory prediction in physics systems. In *International Joint Conference on Neural Networks (IJCNN)*, 2021. 2

[37] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. UniSim: Learning interactive real-world simulators. In *International Conference on Learning Representations (ICLR)*, 2024. 2

[38] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 1

[39] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4

[40] Boyuan Zhao et al. Video as world simulators. *arXiv preprint arXiv:2410.00234*, 2024. 1, 2

Appendix

This appendix provides: (A) complete proofs for all theorems in the main paper; (B) additional experimental results on DMControl, Minecraft, and nuScenes; (C) implementation details; and (D) a discussion of extensions.

A. Complete Proofs

We restate each theorem for completeness before providing the full proof.

A.1. Proof of Theorem 1 (Unitarity)

Theorem 8 (Restatement)**.** *If $f(\cdot, a)$ is measure-preserving w.r.t. μ for all $a \in \mathcal{A}$, then $\mathcal{K}_a$ is a unitary operator on $\mathcal{F} = L^2(\mathcal{X}, \mu)$ with $\|\mathcal{K}_a\| = 1$.*

Proof. We establish the three required properties of a unitary operator.
(i) Linearity. For $\varphi_1, \varphi_2 \in \mathcal{F}$ and scalars $\alpha_1, \alpha_2 \in \mathbb{R}$:

$$\begin{aligned}\mathcal{K}_a(\alpha_1\varphi_1 + \alpha_2\varphi_2)(x) &= (\alpha_1\varphi_1 + \alpha_2\varphi_2)(f(x,a)) \\ &= \alpha_1\varphi_1(f(x,a)) + \alpha_2\varphi_2(f(x,a)) \\ &= \alpha_1(\mathcal{K}_a\varphi_1)(x) + \alpha_2(\mathcal{K}_a\varphi_2)(x).\end{aligned}$$

(ii) Isometry. By the change of variables $y = f(x,a)$ and measure preservation $\mu(f^{-1}(E)) = \mu(E)$:

$$\|\mathcal{K}_a\varphi\|^2 = \int_{\mathcal{X}} |\varphi(f(x,a))|^2 \, d\mu(x) = \int_{\mathcal{X}} |\varphi(y)|^2 \, d\mu(y) = \|\varphi\|^2 .$$

(iii) Surjectivity. For any $\psi \in \mathcal{F}$, since $f(\cdot, a)$ is surjective μ-a.e., define $\varphi(x) := \psi(f^{-1}(x))$ where f^{-1} is the preimage. Then $(\mathcal{K}_a\varphi)(x) = \varphi(f(x,a)) = \psi(x)$, so $\psi \in \mathrm{Im}(\mathcal{K}_a)$.

A linear bijective isometry on a Hilbert space is unitary, and $\|\mathcal{K}_a\| = 1$ follows from the isometry property. □

A.2. Proof of Theorem 2 (Linear Error Accumulation)

Theorem 9 (Restatement)**.** *Under Definition 2, for any action sequence $a_{0:T-1}$:*

$$\left\|\Phi_\theta(x_T) - K_{a_{T-1}} \cdots K_{a_0}\, \Phi_\theta(x_0)\right\|_2 \le T \cdot \varepsilon_{\text{approx}}(d).$$

Proof. Let $z_t = \Phi_\theta(x_t)$ and $\hat{z}_t$ be the model rollout. Define residuals $\delta_t := z_{t+1} - K_{a_t} z_t$, so $\|\delta_t\| \le \varepsilon_{\text{approx}}(d)$ by definition.

Write the error telescopically. For $t = 0$: $z_1 - \hat{z}_1 = (z_1 - K_{a_0} z_0) = \delta_0$. For general T:

$$z_T - \hat{z}_T = \sum_{t=0}^{T-1} \prod_{s=t+1}^{T-1} K_{a_s} \cdot \delta_t,$$

where an empty product (for $t = T-1$) is the identity. This identity follows by induction: at step T,

$$\begin{aligned} z_T - \hat{z}_T &= (K_{a_{T-1}} z_{T-1} + \delta_{T-1}) - K_{a_{T-1}} \hat{z}_{T-1} \\ &= K_{a_{T-1}}(z_{T-1} - \hat{z}_{T-1}) + \delta_{T-1}. \end{aligned}$$

Expanding recursively yields the telescoping sum above. Applying the triangle inequality and $\|K_a\|_2 \le 1$:

$$\|z_T - \hat{z}_T\|_2 \le \sum_{t=0}^{T-1} \underbrace{\left\| \prod_{s=t+1}^{T-1} K_{a_s} \right\|_2}_{\le 1} \cdot \|\delta_t\|_2 \le T \cdot \varepsilon_{\text{approx}}(d). \square$$

A.3. Proof of Theorem 3 (Exponential Accumulation)

Theorem 10 (Restatement)**.** *Let $\hat{f}$ be a transition with Lipschitz constant $L > 1$ and one-step error $\varepsilon > 0$. Then $\|x_T - \hat{x}_T\|_2 \le \varepsilon(L^T - 1)/(L-1)$.*

Proof. Let $e_t = \|x_t - \hat{x}_t\|_2$ with $e_0 = 0$. The one-step recurrence is:

$$\begin{aligned} e_{t+1} &= \left\| f(x_t, a_t) - \hat{f}(\hat{x}_t, a_t) \right\|_2 \\ &\le \left\| f(x_t, a_t) - \hat{f}(x_t, a_t) \right\|_2 + \left\| \hat{f}(x_t, a_t) - \hat{f}(\hat{x}_t, a_t) \right\|_2 \\ &\le \varepsilon + L\, e_t. \end{aligned}$$

This is a linear recurrence $e_{t+1} \le L e_t + \varepsilon$ with $e_0 = 0$. Unrolling:

$$e_T \le \varepsilon \sum_{k=0}^{T-1} L^k = \varepsilon \cdot \frac{L^T - 1}{L - 1}.$$

Since $L > 1$, $\frac{L^T-1}{L-1} \sim L^T/(L-1)$ grows exponentially, while Theorem 2 gives $T\varepsilon$ (linear). □

A.4. Proof of Theorem 4 (Memory Horizon)

Proof. Set the upper bound on prediction error equal to δ and solve for T.
KWM: $T \cdot \varepsilon_{\text{approx}} \le \delta \Rightarrow T \le \delta/\varepsilon_{\text{approx}}$. Take the floor for integer T.
Nonlinear: $\varepsilon(L^T - 1)/(L-1) \le \delta \Rightarrow L^T \le 1 + \delta(L-1)/\varepsilon \Rightarrow T \le \log(1 + \delta(L-1)/\varepsilon)/\log L$.
Asymptotic ratio:

$$\frac{T^*_{\text{KWM}}}{T^*_{\text{naive}}} = \frac{\delta/\varepsilon_{\text{approx}}}{\log(1 + \delta(L-1)/\varepsilon)/\log L} \approx \frac{(\delta/\varepsilon)\log L}{\log(\delta(L-1)/\varepsilon)},$$

which grows without bound as $\delta/\varepsilon \to \infty$. □

A.5. Proofs of Theorems 5, 6, 7

Full proofs of Theorems 5 (necessary embedding dimension), 6 (parallel rollout), and 7 (video consistency bound) are given in the main text and are complete as stated. For Theorem 6, the Blelloch prefix-scan algorithm [3] is standard; we verify the constant factors: each of the $O(\log T)$ rounds performs $O(T/2^k)$ matrix products in round k, so total work is $\sum_{k=1}^{\log T} T/2^k = T(1 - 1/T) = O(T)$, confirming work-optimality.

B. Additional Experimental Results

B.1. DMControl and Minecraft Results

Table 5 reports FVD on DMControl and Minecraft at all horizons. The same qualitative pattern holds as on RoboDesk: KWM achieves linear FVD growth while baselines grow super-linearly. On Minecraft—a more complex environment—the advantage of KWM is even more pronounced at $T = 128$ (4.2× over DreamerV3).

Table 5. **FVD on DMControl and Minecraft.** Bold = best, underlined = second-best.

Dataset	Method	T=16	T=32	T=64	T=128
DMCtrl	DreamerV3	108	175	315	609
	IRIS	<u>102</u>	<u>158</u>	<u>272</u>	<u>484</u>
	Genie	122	208	398	753
	KWM (ours)	**75**	**99**	**130**	**165**
Minecraft	DreamerV3	178	287	518	1001
	IRIS	<u>168</u>	<u>259</u>	<u>447</u>	<u>796</u>
	Genie	201	341	654	1348
	KWM (ours)	**128**	**164**	**213**	**272**

B.2. nuScenes Driving Results

On nuScenes, we additionally report **FID** (image quality of individual frames) alongside FVD. KWM achieves FID = 18.4 at $T = 64$, versus 41.2 for DreamerV3 and 36.8 for IRIS, confirming that the Koopman structure preserves per-frame quality in addition to temporal coherence.

B.3. Sensitivity to Training Horizon

We ablate the training horizon $T_{\text{train}} \in \{8, 16, 32\}$ used in $\mathcal{L}_{\text{pred}}$ (Eq. 1). Increasing T_{train} consistently improves test-time long-horizon FVD but increases training cost. We find $T_{\text{train}} = 16$ offers the best efficiency–performance trade-off, as used in all main experiments.

B.4. Qualitative Rollout Comparison

Qualitative long-horizon rollouts (8-frame strips at $T = 8, 32, 64, 128$) are available in the supplementary video. Visually, DreamerV3 and IRIS begin showing blurring and object drift at $T \geq 32$, while KWM maintains sharp object boundaries and consistent scene layout throughout.

C. Implementation Details

Encoder. ViT-B/16 with patch size 16, image resolution 224×224, pre-trained on ImageNet-21k [7] and fine-tuned. The [CLS] token is projected to $\mathbb{R}^d$ via a linear head. Context stack of $k = 4$ frames is concatenated along the channel dimension.

Koopman matrices. The MLP A_θ has two layers: input size $|\mathcal{A}|$, hidden size 512 (continuous) or $d^2/2$ embedding (discrete), output size d^2. The output is reshaped to $d \times d$ and antisymmetrised: $A = (A_{\text{raw}} - A_{\text{raw}}^\top)/2$. The matrix exponential is computed via the scaled-and-squared Padé approximation [18] (order 13), implemented in PyTorch using `torch.linalg.matrix_exp`.

Decoder. Four upsampling blocks with 3×3 convolutions, GroupNorm, and SiLU activations, from d-dim vector to $3 \times H \times W$. A learned positional grid is added before the first block.

Data augmentation. Random horizontal flipping (probability 0.5), color jitter (brightness, contrast, saturation each ± 0.2), and random cropping to 224×224.

Reproducibility. Code and pretrained model weights will be released at `[URL anonymised for review]`. All experiments use PyTorch 2.3 with CUDA 12.1 on 8×A100-80GB.

D. Discussion: Extensions and Open Problems

Structured Koopman matrices. Block-diagonal parameterisation $K_a = \text{diag}(B_1(a), \ldots, B_{d/b}(a))$ with blocks $B_i \in \mathbb{R}^{b \times b}$ reduces the matrix exponential cost from $O(d^3)$ to $O(d \cdot b^2)$ and the storage from $O(d^2)$ to $O(d \cdot b)$. We leave a systematic study of this to future work.

Stochastic Koopman operators. The current framework assumes deterministic dynamics. For stochastic environments, one can parameterise K_a as a distribution over unitary matrices (via matrix-variate distributions on the unitary group $U(d)$), giving a stochastic Koopman world model analogous to the RSSM in DreamerV3.

Koopman eigenvalue control. Theorem 1 establishes $\rho(K_a) \leq 1$ under the matrix-exponential parameterisation. One can *directly* optimise the Koopman eigenvalue spectrum as part of training (e.g., penalising imaginary-part spread to encourage frequency separation), providing an interpretable spectral structure analogous to Fourier analysis of dynamics.

Connection to linear recurrent models. The Koopman propagation step $z_{t+1} = K_a z_t$ is structurally related to linear recurrent units [29] (LRUs) and selective SSMs [11], which have recently shown strong sequence modelling performance. Our framework provides a dynamical systems justification for the effectiveness of linear recurrences in world modelling and a principled design criterion (unitarity) for their parameterisation.