Flow Matching in Feature Space for Stochastic World Modeling

François Porcher[1], **Nicolas Carion**[1], **Karteek Alahari**[2], **Shizhe Chen**[2]

[1]FAIR at Meta, [2]Inria

World modeling requires forecasting uncertain futures while preserving information useful for downstream perception. Existing visual world models often struggle to satisfy both goals: VAE-based stochastic models operate in low-dimensional reconstruction latents, which can limit perception performance, while deterministic predictors using strong pretrained features collapse multimodal futures into a single blurry mean. In this work, we propose *FlowWM*, a stochastic world model that performs flow matching directly within pretrained feature space (e.g., DINOv3). This is challenging because pretrained features are substantially high-dimensional, making standard diffusion recipes suboptimal. To address this, we investigate the design choices needed for feature-space flow matching and introduce a differentiable one-step projection mechanism that enables efficient training with temporal consistency and task-driven objectives. We evaluate FlowWM on two benchmarks: a synthetic benchmark for systematic evaluation of accuracy and diversity, and a real-world benchmark FuturePerception. FlowWM improves perception performance, mode coverage, and horizon robustness, validating our proposed design for stochastic world modeling in high-dimensional feature spaces.

Date: July 17, 2026
Correspondence: François Porcher at francoisporcher@meta.com
Code: https://github.com/facebookresearch/Flow-World-Models

Meta

1 Introduction

World models aim to forecast future states from past context (Ha and Schmidhuber, 2018). In visual domains like autonomous driving (Vasudevan et al., 2025) and robotics (Wu et al., 2023), effective forecasting requires more than generating visually plausible pixels: the predicted states should preserve information relevant to downstream perception and planning (Maes et al., 2026). This makes the choice of representation central to visual world modeling.

Recent advances in video generative modeling (Liu et al., 2024; Peng et al., 2025; Wan et al., 2025) have driven diffusion (Ho et al., 2020; Song et al., 2021) and flow-matching models (Lipman et al., 2023; Liu et al., 2022) as visual world models. These methods first compress videos into a low-dimensional latent space using a VAE (Alonso et al., 2024), then train neural networks (Peebles and Xie, 2023; Ronneberger et al., 2015) to model the latent dynamics. However, VAE latents are optimized for pixel-wise reconstruction rather than geometry or semantics, and thus are inadequate for perception tasks that are critical for high-level planning such as object localization, detection, and tracking (Kumar et al., 2022; Jayasumana et al., 2024; Yao et al., 2025; Shi et al., 2025).

An alternative is to predict future states in the feature space of pretrained visual encoders such as DINOv3 (Siméoni et al., 2025) and VJEPA-2 (Assran et al., 2025). These representations excel at capturing semantic correspondence and objectness, showing significant promise for perception-oriented future prediction (Zhou et al., 2024; Karypidis et al., 2024). However, current feature-space world models remain largely deterministic, yielding a single future for the given context. This is a critical limitation in complex real-world environments like autonomous driving, where the future is inherently multimodal and stochastic: a vehicle may brake or continue, a pedestrian may cross or stop, and occluded objects may appear or remain hidden. Deterministic predictors trained with standard regression losses tend to average over these possibilities, producing predictions that may correspond to no valid future (see formal proof in Appendix A).

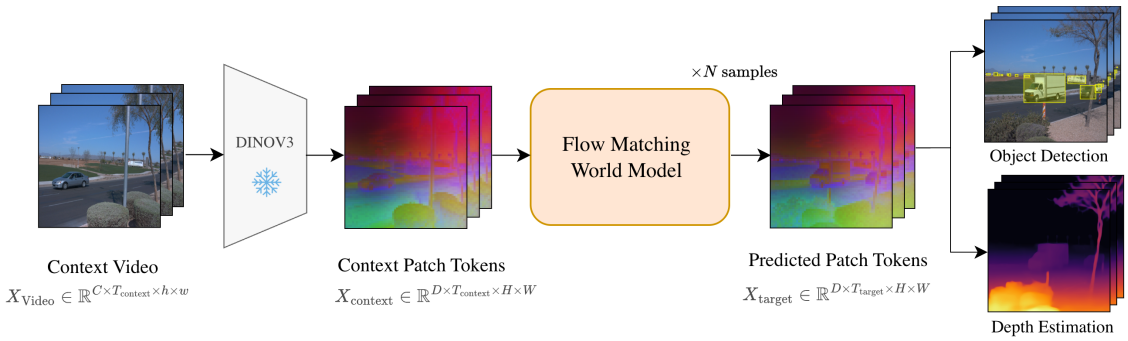


Figure 1 Overview of FlowWM. A frozen DINOv3 backbone encodes context video frames to patch tokens, and our Flow Matching World Model samples possible futures in the same latent space. The predictions are evaluated on perception tasks such as object detection and depth estimation.

To overcome this limitation, recent work (Walker et al., 2025) has explored stochastic world modeling by applying a latent diffusion model to frozen pretrained video features. Yet, adopting the standard DiT recipe (Peebles and Xie, 2023) for low-dimensional VAE latents is unstable or suboptimal in a significantly higher-dimensional space (Mousakhan et al., 2025). Similar challenges have been observed in image generation, where diffusion in pretrained representation spaces requires designs tailored to high-dimensional latents (Zheng et al., 2025). Video forecasting further compounds these difficulties, since generated feature sequences must remain temporally coherent across future frames.

To address these issues, we introduce FlowWM, a stochastic world model that leverages flow matching in high-dimensional feature spaces (e.g., DINOv3 (Siméoni et al., 2025)). We identify and propose the key components needed for feature-space generative modeling, including architectural design, diffusion timestep scheduling, and training objectives for temporal coherence and task relevance. To apply these objectives efficiently, we introduce a differentiable one-step projection mechanism that supervises predicted future features without backpropagating through the full sampling trajectory.

We construct two complementary benchmarks for systematic evaluation: a synthetic stochastic bouncing-shapes environment and a real-world benchmark FuturePerception derived from the Waymo Open Dataset (Sun et al., 2020). The synthetic benchmark provides enumerated multimodal futures, enabling controlled evaluation of prediction accuracy, diversity, and mode coverage. The real-world benchmark evaluates predicted future features through essential downstream perception tasks, including object detection and depth prediction. Across these benchmarks, FlowWM improves perception performance, mode coverage, and horizon robustness compared with deterministic predictors (Zhou et al., 2024) and stochastic models operating in either VAE (Wan et al., 2025) or pretrained feature spaces (Walker et al., 2025). These results demonstrate the effectiveness of the proposed method for stochastic world modeling in high-dimensional feature space.

To summarize, our contributions are as follows:

- We propose FlowWM, a stochastic world model operating directly in pretrained high-dimensional feature spaces.
- We investigate design choices required for stochastic feature-space world models with flow matching, including architecture, timestep scheduling, and efficient temporal consistency and task-driven training via one-step projection.
- We introduce synthetic and real-world benchmarks for evaluation, showing that FlowWM produces state-of-the-art perception performance, mode coverage, and horizon robustness.

We will release the code and benchmarks to accelerate research in this direction.

2 Related Work

World Modeling. World models learn to predict future environment states, enabling imagined rollouts for planning and decision making (Ha and Schmidhuber, 2018). Classical approaches often leverage VAE-based latents (Van Den Oord et al., 2017) within action-conditioned dynamic models (Hafner et al., 2019a,b;

Micheli et al., 2022; Kaiser et al., 2019; Bruce et al., 2024; Gao et al., 2025). While effective in quasi-deterministic settings, they often suffer from representation collapse, making them unsuitable for more stochastic environments (Mousakhan et al., 2025). World models such as DINO-WM (Zhou et al., 2024) and DINO-Foresight (Karypidis et al., 2024) instead predict future states in pretrained visual features (Oquab et al., 2023), demonstrating the benefits of semantically rich representations for robotics and perception. Despite these advances, these models remain largely deterministic (Hansen et al., 2023; Zhou et al., 2024; Karypidis et al., 2024; Assran et al., 2025), limiting their ability to represent uncertainty and multi-modality, particularly over long horizons. Modeling stochastic dynamics in high-dimensional spaces remains challenging (Walker et al., 2025; Mousakhan et al., 2025). Orbis (Mousakhan et al., 2025) avoids this difficulty by distilling DINO features into lower-dimensional VAE latents. The work most closely related to ours is Walker et al. (2025), which applies diffusion directly to frozen visual features for stochastic forecasting. However, its main focus is on comparing frozen representations rather than on improving stochastic world models. In contrast, our work improves stochastic world modeling in high-dimensional latent spaces.

Video Generative Modeling. Video generative modeling learns a video distribution $p(x_{1:T})$, typically with flow matching or diffusion models (Liu et al., 2024; Peng et al., 2025; Wan et al., 2025). Most methods adopt a two-stage pipeline in which a spatiotemporal VAE compresses videos into a low-dimensional latent space, followed by generative modeling in that space (Deng et al., 2025). While this yields high visual fidelity, strong VAE bottlenecks discard fine-grained semantic structure, making the resulting representations poorly suited for dense perception and planning tasks (Karypidis et al., 2024; Jenni and Favaro, 2018; Newell and Deng, 2020). To improve diffusion model training, recent works incorporate task-driven objectives by backpropagating differentiable rewards through the diffusion inference process (Xu et al., 2023; Clark et al., 2024). This requires differentiating through the full or truncated denoising trajectory, which is computationally expensive and prone to vanishing or exploding gradients. Instead, we use a one-step projection that applies downstream supervision at the endpoint, avoiding backpropagation through time while remaining stable and efficient.

Diffusion in High-Dimensional Latent Space. Recent work has shown that diffusion models can operate directly in high-dimensional latent spaces. RAE (Zheng et al., 2025) achieves state-of-the-art image generation by performing diffusion in the latent space of DINO, using a large projection head and a carefully designed time-sampling strategy. Similarly, Just Image Transformers (JIT) (Li and He, 2025) demonstrates that diffusion transformers scale to high-dimensional spaces by projecting directly onto the data manifold rather than predicting a velocity field. While these works focus on image generation, we study stochastic world models, where the model must not only generate plausible future features but also maintain temporal coherence over long prediction horizons.

3 The Proposed Method: FlowWM

3.1 Problem Formulation: Latent Space World Modeling

We aim to train world models *directly in the high-dimensional latent spaces* of pretrained vision encoders, leveraging their strong semantic representations for downstream perception.

Given a video sequence $I_{1:T}$ of raw frames, the first T_{context} frames are encoded frame-by-frame using a frozen pretrained encoder E (e.g., DINOv3 (Siméoni et al., 2025)), and the model must predict the subsequent latent frames of length T_{target}. Formally, we define the encoded context latents, the feature-space representations of observed frames, as $x_{\text{ctx}} = x_{1:T_{\text{context}}} = E(I_{1:T_{\text{context}}})$, where $x_{\text{ctx}} \in \mathbb{R}^{T_{\text{context}} \times H \times W \times D}$ with H, W, D denoting the spatial height, width and channel dimension of the latent feature map, respectively. The objective is to generate future latents $\hat{x}_{\text{future}} = \hat{x}_{T_{\text{context}}+1:T} \in \mathbb{R}^{T_{\text{target}} \times H \times W \times D}$.

Evaluation is performed on downstream *perception tasks* using pretrained, frozen decoders applied to the predicted features. This evaluation explicitly probes the preservation of object-centric information and scene geometry which are essential for planning and control, whereas low-level reconstruction metrics that capture pixel- or feature-level statistics may not correlate with task utility.

3.2 Stochastic World Model with Flow Matching

To address the limitations of deterministic latent world models, we introduce FlowWM, a stochastic latent space world model with flow matching, as illustrated in Figure 1.

Flow Matching. We use flow matching (Lipman et al., 2023; Liu et al., 2022) with the standard linear probability path. Let x_0 denote a noise latent and x_1 a data latent corresponding to future frames conditioned on a context latent x_{ctx}. We use $\tau \in (0, 1)$ to denote the continuous interpolation variable along the probability path[1], and a linear interpolation between noise and data is computed as:

$$x_\tau = (1 - \tau)\, x_0 + \tau\, x_1, \qquad \tau \in [0, 1]. \tag{1}$$

For this linear path, the ground-truth velocity field is $u^\star(x_\tau, \tau) = x_1 - x_0$. A neural network $u_\theta(x_\tau, x_{\text{ctx}}, \tau)$ is trained to approximate this velocity, thereby learning the conditional distribution $p(x_{\text{future}}|x_{\text{ctx}})$. Given a trained velocity field, sampling is performed by integrating the ODE $\frac{dx_\tau}{d\tau} = u_\theta(x_\tau, x_{\text{ctx}}, \tau)$, starting from $x_0 \sim p_{\text{noise}}$ and integrating from $\tau = 0$ to $\tau = 1$ to obtain a final prediction $\hat{x}_1$.

Model Architecture. We adopt a Transformer-based latent generative architecture operating directly in pretrained semantic representations (Figure 2). The model takes a noisy target latent x_τ as input and is conditioned on encoded context frames x_{ctx} via cross-attention. We decompose positional embeddings into temporal and spatial components and apply rotary positional encodings (RoPE) (Su et al., 2024) to both x_τ and x_{ctx}.

The model consists of a backbone for multimodal fusion and a projection head for velocity prediction. The backbone contains 2 DiT-style Transformer blocks with dimension 256, using query–key normalization (Henry et al., 2020) for stability. The resulting features are fed through AdaLN conditioning into a shallow but wide projection head (Zheng et al., 2025) composed of 2 wide projection layers (projection dimension 1024), which predicts the flow field $u_\theta(x_\tau, \tau)$. The wide projection dimension (1024) ensures the head's hidden dimension exceeds the per-patch latent dimension (d=384), which is critical for accurately predicting velocity fields in high-dimensional latent spaces.

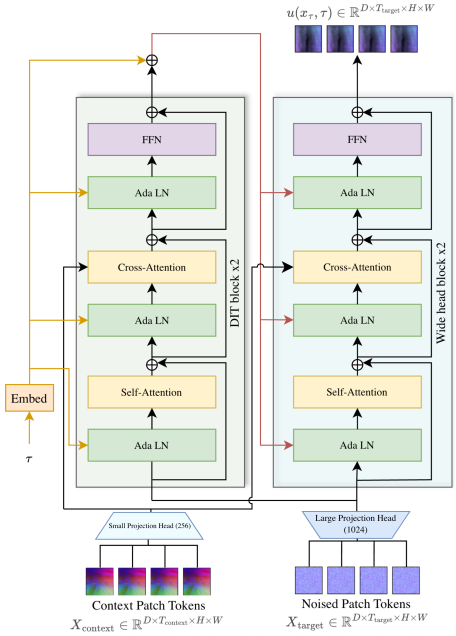


Figure 2 Overview of our FlowWM architecture. The model takes as input a noisy target latent x_τ, conditioned on encoded context frames x_{ctx}, to predict the flow field $u_\theta(x_\tau, \tau)$.

3.3 Training Objectives

Standard Flow Matching Objective. We optimize the model using the conditional expected squared error (Lipman et al., 2023):

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{\substack{x_1 \sim p_{\text{data}},\, x_0 \sim p_{\text{noise}}, \\ \tau \sim \mathcal{U}(0,1)}} \left[\left\| u_\theta(x_\tau, \tau) - (x_1 - x_0) \right\|_2^2 \right]. \tag{2}$$

Time Consistency with One-Step Projection. Video generation requires *temporal consistency*, ensuring that dynamic entities adhere to plausible motion trajectories. Achieving such consistency remains a central challenge, especially as the temporal horizon increases, with existing models exhibiting artifacts like flickering or inconsistent motions (Ho et al., 2022; Yan et al., 2023).

To address this, we propose a regularization term inspired by Physics-Informed Neural Networks (Raissi et al., 2019) that constrains not only the model output, but also its derivatives. Specifically, we regularize predicted latents to respect the time derivative of ground-truth latents.

Directly imposing losses on the generated endpoint x_1 requires integrating the ODE, which is computationally expensive and prone to vanishing or exploding gradients (Hu et al., 2021; Xu et al., 2023). Instead, we propose a *one-step projection* to obtain a differentiable estimate of the predicted endpoint x_1 without integrating the full flow.

[1]We reserve t exclusively for the model's input time coordinate to avoid confusion with τ.

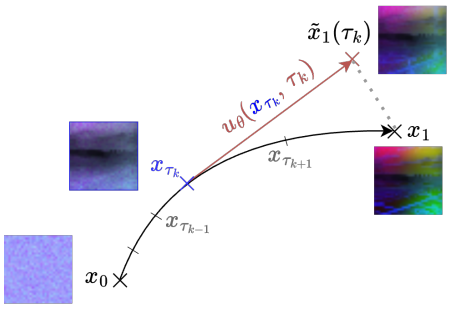


Figure 3 One-step projection in Flow Matching. A differentiable estimate of the final endpoint is obtained from x_τ by adding the predicted velocity, yielding $\tilde{x}_1(\tau)$ without integrating the full flow.

Given x_τ at time τ, we approximate the denoised endpoint via a linear projection along the current velocity vector u_θ:

$$\tilde{x}_1(\tau) = x_\tau + (1-\tau)\, u_\theta(x_\tau, \tau). \tag{3}$$

Using this one-step projection, we impose temporal consistency directly on the projected endpoint. Let x_t^{GT} denote the ground-truth latents of the frame at time t. We define the ground-truth temporal derivative as:

$$\Delta x_t^{\mathrm{GT}} = x_{t+1}^{\mathrm{GT}} - x_t^{\mathrm{GT}}. \tag{4}$$

We then enforce alignment between the ground-truth temporal differences and those of the predicted latents obtained via one step projection Eq. (3), yielding an auxiliary loss:

$$\mathcal{L}_{\mathrm{temporal}} = \sum_{t=1}^{T-1} \left\| \Delta\tilde{x}_t - \Delta x_t^{\mathrm{GT}} \right\|^2, \tag{5}$$

where $\tilde{x}_{1:T}$ are the projected endpoints of video frames. This encourages the predicted latents at the end of the flow to have consistent temporal derivatives as real latents.

Task-driven Objective. The flow matching and temporal consistency objectives enable training general latent world models without any task-specific supervision. A natural question then arises: *if the downstream task is known at training time, can it provide additional supervision to further shape the world model?*

The one-step projection introduced in Eq. (3) provides an efficient and stable mechanism to incorporate task-driven objectives (e.g., manifold regularization, differentiable rewards, or detector-based losses) into flow-matching training.

Assume a downstream perception task such as object detection with a pretrained object detector operating in the latent space. We first leverage the one-step projection to obtain a differentiable estimate of the final predicted latent state, denoted by $\tilde{x}_1$. The frozen detector is then applied to this projected endpoint, and its detection loss $\mathcal{L}_{\mathrm{det}}(\tilde{x}_1)$ is backpropagated only through the projection and the world model. In this way, downstream task supervision actively steers the world model towards representations that are more semantically useful for the downstream application.

3.4 Shifting of Timestep Schedules

When increasing the dimension of the tensor being generated, both the timestep sampling distribution during training and the inference scheduler must be adjusted to maintain a stable signal-to-noise ratio.

We use the resolution-dependent timestep shift of Esser et al. (2024), which rescales diffusion time by a factor α, toward noisier timesteps. Although this strategy was initially introduced for generating images in higher resolution, the same principle applies when scaling latent dimension or prediction horizon, since both increase the number of noise dimensions. Specifically, the original timestep $\tau \in [0, 1]$ is transformed as

$$\tau' = \frac{\alpha\,\tau}{1 + (\alpha - 1)\,\tau}, \tag{6}$$

which preserves the endpoints while redistributing probability mass toward larger noise levels.

4 Latent World Model Benchmarks

To systematically benchmark latent world models, particularly focusing on perception evaluation which is more relevant for planning and control than pixel-wise generation, we introduce two complementary benchmarks spanning synthetic and real-world environments. The synthetic benchmark provides controlled stochastic settings with simple objects, enabling comprehensive and rigorous evaluation of model quality. In contrast, our FuturePerception benchmark presents richer visual complexity and dynamics, but offers less rigid evaluation due to limited ground-truth structure.

4.1 Synthetic Benchmark: Bouncing Shapes

The *Bouncing Shapes* benchmark defines a controlled stochastic environment in which two objects (a red square and a blue ball) move inside a 2D box and bounce off the walls. At each wall collision, an object either performs a standard bounce or reverses its incoming velocity with probability 0.5, inducing an exponential branching of possible futures. We can enumerate *all* futures consistent with the observed context. We use 16 context frames to predict the next 16 frames.

Evaluation metrics. We evaluate the future latent prediction quality via a pretrained decoder that predicts the 2D coordinates of the objects from the latent space. Since all possible futures can be enumerated, we can measure metrics that are not available in real datasets:

- *Precision Error:* Do sampled predictions correspond to valid modes? Each prediction is matched to its nearest ground-truth mode, and the average matching error over all predictions is reported.
- *Recall Error:* Does the model cover all valid modes? Each ground-truth mode is matched to its nearest sampled prediction, and the average matching error over all ground-truth modes is reported.

We also define an F1 error as the harmonic mean of Precision Error and Recall Error. The details of the benchmark and metrics are provided in Appendix B.

4.2 Real-world Benchmark: FuturePerception

We construct our real-world benchmark based on the Waymo Open Dataset (Sun et al., 2020), originally introduced for object tracking. Although Walker et al. (2025) proposed a similar protocol, no code or splits were released, so we recreate our FuturePerception task on top of Waymo. No data filtering is applied, so the dataset preserves the natural distribution of real-world autonomous driving scenarios. The task uses 4 context frames to predict the next 12 frames, which corresponds to 1.2 seconds into the future at 10 FPS. Examples are provided in Appendix C.

The FuturePerception task is challenging for three main reasons: i) *Resolution.* Waymo videos are high-resolution (1920×1280), making diffusion- and flow-based video modeling substantially more difficult than on ImageNet-scale datasets (256×256). ii) *Dynamics.* Real-world driving scenes involve many interacting agents with complex motion patterns, including occlusions and viewpoint changes induced by ego-motion (e.g., vehicles passing behind others or pedestrians temporarily disappearing). iii) *Stochasticity and Partial Observability.* The future is inherently multi-modal and partially observed. Given the same context, agents may take different plausible actions (e.g., braking or continuing), while new objects may enter the scene or previously occluded ones may reappear, requiring the model to represent uncertainty rather than commit to a single deterministic outcome.

Evaluation metrics. The latent prediction quality is evaluated in downstream object detection and depth prediction tasks. For object detection, we focus on three classes: vehicle, pedestrian, and cyclist, and restrict the evaluation to large objects, for which detection is reliable at this resolution.

We evaluate detection performance using $\mathbf{AP_L(N)}$, which reports the maximum $\mathrm{AP_L}$ across N sampled predictions. This best-of-N metric captures the model's capacity to represent multi-modal futures, rather than averaging over incompatible outcomes.

For depth prediction, we report standard monocular depth metrics (Eigen et al., 2014): RMSE and threshold accuracy δ_i (fraction of pixels where $\max(d/d^*, d^*/d) < 1.25^i$, for $i \in \{1, 2, 3\}$).

5 Experiments

5.1 Experimental Setups

Baselines. We compare FlowWM against the following three classes of baselines:

- Deterministic world models using high-dimensional latents: we consider DINO-WM (Zhou et al., 2024) which predicts future latents *autoregressively*, and a stronger deterministic baseline that predicts the entire future latent trajectory *jointly*, using the same architecture as FlowWM for a fairer comparison. The DINOv3 encoder ($D = 384$) is used for both models.

- Stochastic world models using low-dimensional VAE latents: we replace the DINOv3 encoder with the VAE encoder used in WAN 2.2 (Wan et al., 2025), which compresses each frame into a low-dimensional latent representation ($D = 16$).
- State-of-the-art stochastic world models using high-dimensional latents: we re-implement the model (Walker et al., 2025) with DINOv3 features following discussions with the authors.

Implementation Details. We use a frozen DINOv3 ViT-S model (Siméoni et al., 2025) as encoder E and encode each frame separately. We average pool token features from every third layer (3, 6, 9, 12), motivated by findings that intermediate layers can provide stronger representations (Bolya et al., 2025). We train all models using AdamW with a learning rate of 10^{-3} and an effective batch size of 128 for 52K gradient steps (150 epochs total). We apply gradient clipping with a 1.0 norm, use a linear warmup for 10 epochs, and cosine annealing afterwards. Images from the Waymo dataset are resized to 512×512, and are augmented with random horizontal flip. The model is trained with 64 NVIDIA V100 GPUs in fp32 precision, taking roughly 25 hours. For inference, we use the Euler solver with sampling over 50 steps. For object detectors, we use the DINO-DETR architecture (Zhang et al., 2022) based on DETR (Carion et al., 2020), implemented in the detrex codebase (Ren et al., 2023) and relying on detectron2 (Wu et al., 2019). For depth prediction, we train a lightweight depth head on DINOv3/WAN VAE latents using pseudo-labels from Depth Anything V2 (Yang et al., 2024).

Table 1 Results on the Bouncing Shapes benchmark. TC denotes temporal consistency loss. We report precision, recall, and F1 errors, where lower is better. The Oracle uses ground-truth future latents. Lower precision error indicates more accurate samples; lower recall error indicates greater diversity and mode coverage.

Type	Prec. ↓	Rec. ↓	F1 ↓
Dino-WM	16.1	19.9	17.8
Deterministic Predictor	15.7	18.8	17.1
Walker et al. (2025)	14.5	14.3	14.4
FlowWM w/o TC	4.57	4.49	4.53
FlowWM (Ours)	**4.35**	**4.28**	**4.31**
Oracle	*1.01*	*0.98*	*1.00*

5.2 Results on the Synthetic Benchmark

Table 1 reports performance of deterministic and stochastic world models using the same high-dimensional DINOv3 features on our synthetic benchmark. As the task is relatively simple, we do not apply the task-driven objective to train our model. For reference, we report the Oracle performance of evaluating the coordinate predictor on ground-truth future latents.

FlowWM consistently outperforms deterministic predictors, including DINO-WM and our deterministic variant with the same architecture, across all evaluation metrics. This confirms the importance of explicitly modeling stochasticity in multimodal future prediction. The stochastic baseline of Walker et al. (2025) improves over deterministic models, but remains substantially worse than FlowWM, suggesting that standard flow-matching designs are suboptimal in high-dimensional feature spaces. In contrast, FlowWM sharply reduces the F1 error from 14.4 to 4.53 even without temporal consistency. Adding the temporal consistency loss further improves the F1 error to 4.31. These gains come from improvements in both precision and recall: sampled trajectories are closer to valid futures, while the model also covers the set of plausible futures more effectively.

5.3 Results on the Real-World Benchmark

We further evaluate scalability of our approach on the more challenging and realistic FuturePerception benchmark on downstream object detection and depth estimation tasks.

Object detection. As shown in Table 2, FlowWM substantially outperforms both deterministic and VAE-based world models. The gap to deterministic baselines confirms that stochastic modeling remains critical in real-world, high-dimensional settings. Moreover, models operating in VAE latent spaces perform worse than

Table 2 Object detection and depth estimation performance on FuturePerception.

Method	Backbone	Object Detection		Depth Prediction			
		$AP_L(3)\uparrow$	$AP_L(6)\uparrow$	RMSE$\downarrow$	$\delta_1\uparrow$	$\delta_2\uparrow$	$\delta_3\uparrow$
DINO-WM	DINOv3	14.5	14.5	0.12	0.51	0.76	0.88
Deterministic Predictor	DINOv3	15.2	15.2	0.11	0.54	0.78	0.89
WAN 2.2 VAE WM	Wan 2.2 VAE	17.5	18.2	0.10	0.57	0.80	0.90
WAN 2.2 VAE WM (re-encoded)	DINOv3	16.5	17.0	0.107	0.54	0.78	0.89
Walker et al. (2025)	DINOv3	16.5	17.1	0.102	0.56	0.79	0.90
FlowWM (Ours)	DINOv3	**20.9**	**21.7**	**0.078**	**0.723**	**0.858**	**0.913**
Oracle	DINOv3	*65.1*	*65.1*	*0.043*	*0.854*	*0.951*	*0.977*

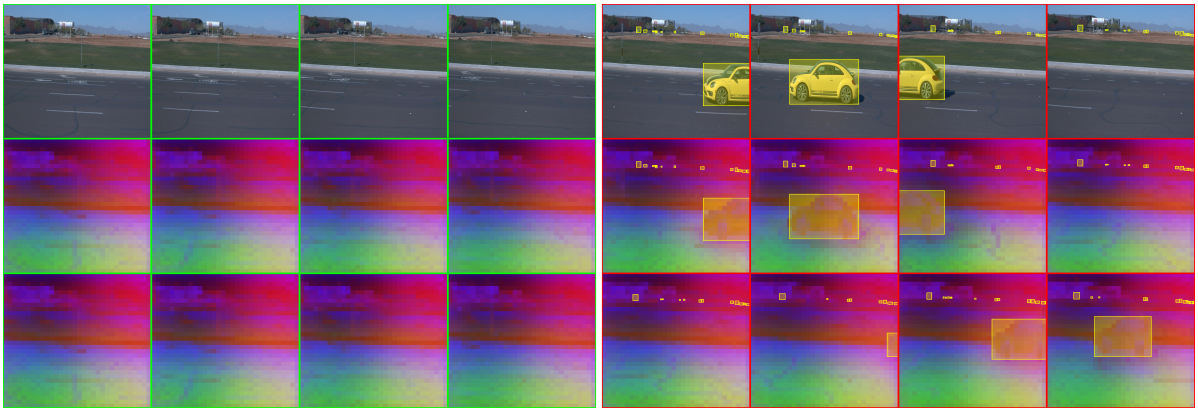


Figure 4 Qualitative examples on our FuturePerception benchmark. Top row: Ground truth frames and oracle detections. Middle and bottom rows: PCA visualizations of predicted latents with overlaid detections for two distinct rollouts. The middle row shows the model anticipating an oncoming vehicle. The bottom row depicts a prediction where a different vehicle appears later. Green borders indicate context frames; red borders indicate predicted future frames.

ours using strong semantic latents. We attribute this to the limited semantic fidelity of low-dimensional VAE representations, which are aligned worse with downstream perception tasks such as detection. To disentangle whether the VAE baseline underperforms due to weaker future modeling or a less semantic feature space, we re-encode VAE predictions into DINOv3 space before detection (VAE decode $\rightarrow$ DINOv3 encode). Re-encoding does not help and slightly hurts performance (17.5 $\rightarrow$ 16.5 $AP_L(3)$), because the pixel roundtrip introduces reconstruction artifacts that degrade the DINOv3 features. This confirms that the information is already lost at the prediction stage, and no post-hoc re-encoding can recover it. FlowWM also outperforms the prior stochastic world model Walker et al. (2025), validating the effectiveness of the proposed design.

Depth estimation. In the right block of Table 2, we further extend our evaluation to depth estimation, another downstream perception task, to assess the generality of the predicted latents. Again, FlowWM outperforms all baselines across all depth metrics, confirming that the predicted latents remain effective across diverse downstream tasks.

Pixel generation. We additionally evaluate the quality of decoded future frames using Fréchet Video Distance (FVD) (Unterthiner et al., 2018) (the lower the better), although pixel generation is not the primary focus of this work. To this end, we train a lightweight pixel decoder from DINOv3 latents and compute FVD on the decoded future frames. FlowWM achieves an FVD of 87.3, corresponding to a 43% improvement over the deterministic predictor (152.4), confirming that our predicted futures are distributionally closer to real driving videos.

Oracle comparison. For reference, we also include an Oracle that applies the downstream predictors to ground-truth future latents. The remaining gap between FlowWM and the Oracle highlights the inherent difficulty of real-world future prediction. Nevertheless, qualitative results in Figure 4 show that FlowWM produces diverse and plausible futures.

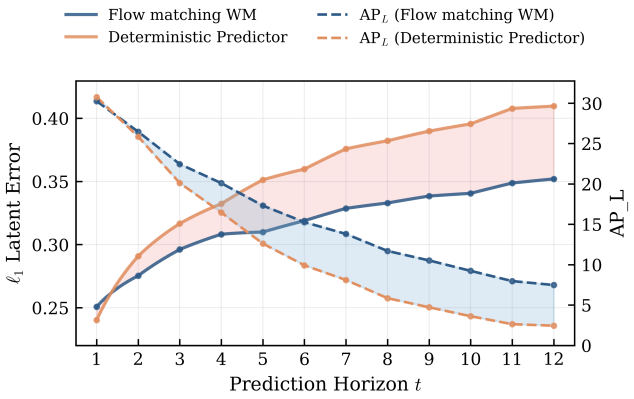


Figure 6 ℓ_1 latent error and AP$_L$ as a function of prediction horizon. Deterministic predictors exhibit faster error growth and AP$_L$ degradation, while our stochastic model maintains lower error and higher performance over long horizons.

5.4 Ablations

We perform extensive ablations on FUTUREPERCEPTION to verify the key design choices of our FlowWM model. Figure 5 shows the overall roadmap of key ingredients, while more detailed results are presented in Appendix D.

Depth and width scaling. We study the effect of model depth and width in latent world models. Varying model depth from 2 to 8 DiT layers (Figure 15a) yields no improvement in downstream detection performance, indicating that model depth is not a limiting factor for the task. In contrast, varying the width of the projection head (Figure 15b) has a pronounced effect: narrow heads severely underperform, while increasing width yields improvements up to our chosen depth $d = 1024$, beyond which gains saturate. This suggests that the width should be at least as large as the latent dimension, aligning with findings in the image domain (Zheng et al., 2025).

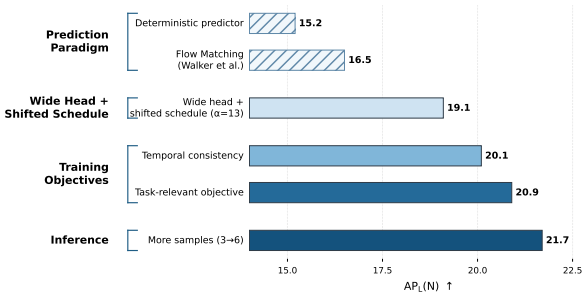


Figure 5 Roadmap for FlowWM, measured by downstream object detection performance on FUTUREPERCEPTION benchmark.

Temporal consistency. In Table 3, we ablate the temporal consistency loss (Eq. (5)). It improves by 5.2% relatively to the baseline, indicating that enforcing coherent temporal dynamics benefits downstream perception. Moreover, incorporating this loss via the proposed one-step projection incurs no additional computational overhead during training.

Task-driven objective. We also ablate the downstream task-driven objective, in which the detector loss is backpropagated during training on the FUTUREPERCEPTION benchmark. This provides additional performance gains. We hypothesize that these gains are bounded by the limited robustness of the frozen detector to noisy predicted latents, which results in weaker gradients. Moreover, because the quality of the projected endpoint latent varies across flow-matching timesteps, appropriately weighting the task-driven loss over timesteps is important (Table 4).

Shifting of timestep schedule. In Table 5, we show that skewing the training timestep distribution (Section 3.4) toward noisier timesteps is crucial. While this strategy was originally introduced to facilitate high-resolution image generation (Peebles and Xie, 2023), our results reveal that the same principle applies when increasing latent dimension and prediction horizon in video world modeling.

Scaling with number of samples. One key advantage of stochastic world models is the ability to sample multiple futures. We study how detection performance scales with the number of sampled futures N. Figure 16 shows **AP$_L$(N)** as a function of the sampling budget. The performance improves consistently as N increases. In contrast, deterministic world models produce the same future on the same context, and thus provide no benefit from additional samples.

5.5 Temporal Error Analysis

As deterministic predictors learn to predict the mean of possible futures, while short-term predictions can be accurate, they tend to degrade quickly as uncertainty compounds over time. In contrast, stochastic world models can represent multiple plausible futures and are therefore expected to degrade more slowly as the forecasting horizon grows.

To test this, we analyze how prediction performance evolves as a function of the number of predicted frames. Figure 6 reports the ℓ_1 latent error across the prediction horizon. For the stochastic model, we sample 3 trajectories and report the performance of the closest to the ground-truth, similar to our detection metric. As expected, both models exhibit increasing error over time, but the deterministic predictor degrades much faster, as it strays further from a valid mode, while the stochastic one keeps predicting a valid future. The figure also shows that the latent reconstruction error directly correlates with the evolution of the detection performance $\mathrm{AP_L(N)}$, which exhibits a similar widening gap over time.

These results demonstrate that deterministic world models accumulate error more rapidly in latent space, which in turn degrades downstream task performance. Stochastic world models mitigate this failure mode by preserving diverse future hypotheses, resulting in more stable and useful predictions over long horizons.

6 Conclusion

This work shows that accurate future prediction in latent space requires both stochastic modeling and semantically rich representations. Deterministic predictors collapse multimodal futures and degrade quickly over long horizons, and low-dimensional VAE latents lack the semantic structure needed for downstream perception. To address this, we propose FlowWM, a stochastic Flow Matching model which operates directly in pretrained visual features and benefits from temporal and task-driven objectives. We further introduce a synthetic benchmark and a real-world FuturePerception benchmark. Our model achieves better mode coverage, slower horizon degradation, and stronger downstream detection performance, highlighting the importance of the proposed design choices. Discussions on limitation and broader impact are presented in the Appendix J and K.

Acknowledgements

S. Chen was funded in part by the French government under management of Agence Nationale de la Recherche as part of the "France 2030" program, reference ANR-23-IACL-0008 (PR[AI]RIE-PSAI projet) and the ANR project 3D-GEM (ANR-25-CE23-7777-01). K. Alahari was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean Government (MSIT) (No. RS-2024-00457882, National AI Research Lab Project).

References

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. In *Thirty-eighth Conference on Neural Information Processing Systems*, 2024. https://arxiv.org/abs/2405.12399.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Rasheed, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Daniel Li, Piotr Dollár, and Christoph Feichtenhofer. Perception encoder: The best visual embeddings are not at the output of the network, 2025. https://arxiv.org/abs/2504.13181.

Jake Bruce, Michael D 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 *Forty-first International Conference on Machine Learning*, 2024.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers, 2020. https://arxiv.org/abs/2005.12872.

Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards, 2024. https://arxiv.org/abs/2309.17400.

Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization, 2025. https://arxiv.org/abs/2412.14169.

David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27, 2014.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. https://arxiv.org/abs/2403.03206.

Shenyuan Gao, Siyuan Zhou, Yilun Du, Jun Zhang, and Chuang Gan. Adaworld: Learning adaptable world models with latent actions. *arXiv preprint arXiv:2503.18938*, 2025.

David Ha and Jürgen Schmidhuber. World models. 2018. doi: 10.5281/ZENODO.1207631. https://zenodo.org/record/1207631.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019b.

Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. *arXiv preprint arXiv:2310.16828*, 2023.

Zhe He, Tianjia Xiao, Shilong Liu, Xinggang Wang, and Long Jiang. Reward feedback learning for latent diffusion models, 2023. https://arxiv.org/abs/2304.05977.

Alex Henry, Prudhvi Raj Dachapally, Shubham Shantaram Pawar, and Yuxuan Chen. Query-key normalization for transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4246–4253, 2020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models, 2022. https://arxiv.org/abs/2204.03458.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. https://arxiv.org/abs/2106.09685.

Sadeep Jayasumana, Srikumar Ramalingam, Andreas Veit, Daniel Glasner, Ayan Chakrabarti, and Sanjiv Kumar. Rethinking fid: Towards a better evaluation metric for image generation. *arXiv preprint arXiv:2401.09603*, 2024. https://arxiv.org/abs/2401.09603. Revised version (v2), submitted Jan 2024.

Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, et al. Model-based reinforcement learning for atari. *arXiv preprint arXiv:1903.00374*, 2019.

Efstathios Karypidis, Ioannis Kakogeorgiou, Spyros Gidaris, and Nikos Komodakis. Dino-foresight: Looking into the future with dino, 2024. https://arxiv.org/abs/2412.11673.

Abhishek Kumar, Rishabh Sinha, Ben Poole, and Diederik P. Kingma. On the sharpness of variational autoencoders. *arXiv:2209.06838*, 2022.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise, 2025. https://arxiv.org/abs/2511.13720.

Yaron Lipman, Ricky T. Q. Chen, Haggai Ben-Hamu, and Maximilian Nickel. Flow matching for generative modeling. In *ICLR*, 2023.

Xuechen Liu, Mingyuan Gong, Qiang Li, Bennett Rue, and Jiaming Song. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv:2209.03003*, 2022.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, Lifang He, and Lichao Sun. Sora: A review on background, technology, limitations, and opportunities of large vision models, 2024. https://arxiv.org/abs/2402.17177.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworldmodel: Stable end-to-end joint-embedding predictive architecture from pixels. *arXiv preprint arXiv:2603.19312*, 2026.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. *arXiv preprint arXiv:2209.00588*, 2022.

Arian Mousakhan, Sudhanshu Mittal, Silvio Galesso, Karim Farid, and Thomas Brox. Orbis: Overcoming challenges of long-horizon prediction in driving world models. *arXiv preprint arXiv:2507.13162*, 2025.

Alejandro Newell and Jia Deng. How useful is self-supervised pretraining for visual tasks? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7345–7354, 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023.

Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, Yuhui Wang, Anbang Ye, Gang Ren, Qianran Ma, Wanying Liang, Xiang Lian, Xiwen Wu, Yuting Zhong, Zhuangyan Li, Chaoyu Gong, Guojun Lei, Leijun Cheng, Limin Zhang, Minghao Li, Ruijie Zhang, Silan Hu, Shijie Huang, Xiaokang Wang, Yuanheng Zhao, Yuqi Wang, Ziang Wei, and Yang You. Open-sora 2.0: Training a commercial-level video generation model in 200k, 2025. https://arxiv.org/abs/2503.09642.

Mihir Prabhudesai, Anirudh Goyal, Deepak Pathak, and Katerina Fragkiadaki. Aligning text-to-image diffusion models with reward backpropagation, 2023. https://openreview.net/forum?id=Vaf4sIrRUC. Submitted to ICLR 2024.

Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi: 10.1016/j.jcp.2018.10.045.

Tianhe Ren, Shilong Liu, Feng Li, Hao Zhang, Ailing Zeng, Jie Yang, Xingyu Liao, Ding Jia, Hongyang Li, He Cao, Jianan Wang, Zhaoyang Zeng, Xianbiao Qi, Yuhui Yuan, Jianwei Yang, and Lei Zhang. detrex: Benchmarking detection transformers, 2023.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Minglei Shi, Haolin Wang, Wenzhao Zheng, Ziyang Yuan, Xiaoshi Wu, Xintao Wang, Pengfei Wan, Jie Zhou, and Jiwen Lu. Latent diffusion model without variational autoencoder, 2025. https://arxiv.org/abs/2510.15301.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025. https://arxiv.org/abs/2508.10104.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *ICLR*, 2021.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

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

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

Arun Balajee Vasudevan, Neehar Peri, Jeff Schneider, and Deva Ramanan. Planning with adaptive world models for autonomous driving. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14938–14945. IEEE, 2025.

Jacob Walker, Féliz Vélez, Ekin Cabrera, Chenxi Zhou, Rishabh Kabra, Carl Doersch, Maks Ovsjanikov, João Carreira, and Shiry Ginosar. Generalist forecasting with frozen video models via latent diffusion. *arXiv preprint arXiv:2507.13942*, 2025. https://arxiv.org/abs/2507.13942.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models, 2025. https://arxiv.org/abs/2503.20314.

Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *Conference on robot learning*, pages 2226–2240. PMLR, 2023.

Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation, 2023. https://arxiv.org/abs/2304.05977.

Wilson Yan, Danijar Hafner, Stephen James, and Pieter Abbeel. Temporally consistent transformers for video generation, 2023. https://arxiv.org/abs/2210.02396.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv preprint arXiv:2406.09414*, 2024.

Jingfeng Yao, Bin Yang, and Xinggang Wang. Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models. *arXiv preprint arXiv:2501.01423*, 2025. doi: 10.48550/arXiv.2501.01423. https://arxiv.org/abs/2501.01423.

Hao Zhang, Feng Li, Shilong Liu, Lei Zhang, Hang Su, Jun Zhu, Lionel M. Ni, and Heung-Yeung Shum. Dino: Detr with improved denoising anchor boxes for end-to-end object detection, 2022. https://arxiv.org/abs/2203.03605.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025. https://arxiv.org/abs/2510.11690.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning, 2024. https://arxiv.org/abs/2411.04983.

Appendix

A Multimodal Futures Require Stochastic Models

Future states are often intrinsically multimodal: agents may take different actions, and objects may move differently due to environmental uncertainty. As a result, a core requirement of world models is to capture multiple plausible futures conditioned on the same past. However, most existing latent-space based world models (Zhou et al., 2024; Karypidis et al., 2024) rely on deterministic predictors trained with pointwise regression losses, which fundamentally cannot represent multimodal dynamics.

Consider a simple bimodal future: conditioned on the same context, the system evolves according to a fair coin toss:

$$Y = \begin{cases} y_L, & \text{with probability } \frac{1}{2}, \\ y_R, & \text{with probability } \frac{1}{2}, \end{cases} \qquad y_L \neq y_R, \tag{7}$$

A deterministic predictor trained with an ℓ_2 loss outputs the conditional mean $\mathbb{E}[Y \mid X_t = x_t] = \frac{1}{2}(y_L + y_R)$, which lies between modes and corresponds to no valid future (Figure 7).

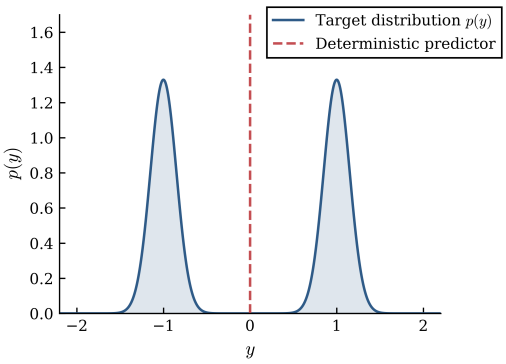


Figure 7 In a bimodal target distribution, deterministic predictors trained with ℓ_1 or ℓ_2 losses collapse to a single point estimate (the conditional mean or median), which lies between modes and corresponds to no valid future.

More generally, deterministic world models trained with ℓ_1 or ℓ_2 losses converge to the conditional mean or median, averaging over all plausible futures rather than representing any valid mode of the distribution.

Formal analysis. Deterministic predictors trained with an ℓ_1 or ℓ_2 loss solve a *conditional regression* problem. Given a context $X_t = x_t$, the model outputs a single value $\hat{y}(x_t)$ and minimizes a risk functional. We analyze both cases below for the bimodal distribution where $Y = y_L$ or $Y = y_R$ each with probability $\frac{1}{2}$.

Squared loss (ℓ_2). The ℓ_2 risk is

$$R_2(\hat{y}) = \tfrac{1}{2}(y_L - \hat{y})^2 + \tfrac{1}{2}(y_R - \hat{y})^2.$$

Setting the derivative to zero yields

$$\hat{y}^\star = \tfrac{1}{2}(y_L + y_R).$$

More generally, the minimizer of the squared loss $\mathcal{R}(\hat{y}) = \mathbb{E}[\|\hat{y}(X_t) - Y\|_2^2]$ is the conditional expectation:

$$\hat{y}^\star(x_t) = \mathbb{E}[\, Y \mid X_t = x_t \,].$$

Thus, any deterministic predictor returns a *single* point estimate equal to the conditional mean over all plausible futures.

Absolute loss (ℓ_1). Similarly, the ℓ_1 risk is

$$R_1(\hat{y}) = \tfrac{1}{2}|y_L - \hat{y}| + \tfrac{1}{2}|y_R - \hat{y}|,$$

whose minimizers are the medians of Y; in particular, the midpoint $\frac{1}{2}(y_L + y_R)$ is always optimal.

In both cases, the predictor returns a single point estimate that averages over all plausible futures, yielding an output that does not correspond to any valid mode of the distribution.

This failure mode becomes worse in *high-dimensional latent space world models*, where averaging destroys semantic structure and temporal compounding amplifies errors. Thus, stochastic generative modeling is essential for robust world modeling.

B Bouncing Shapes Benchmark

B.1 Bouncing Shapes Modularity

The dataset is fully editable and can be made arbitrarily complex. It is possible to change the number, color, shapes of objects, and use arbitrarily complex distributions for rebounds and trajectories. Figure 8 and 9

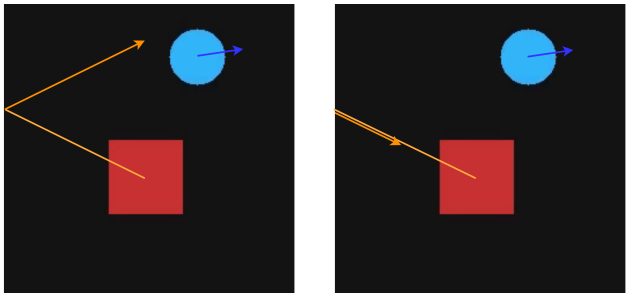
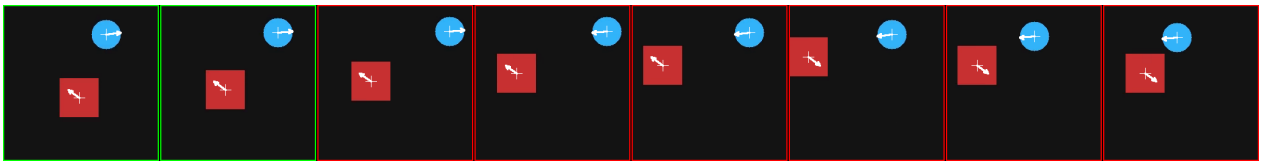
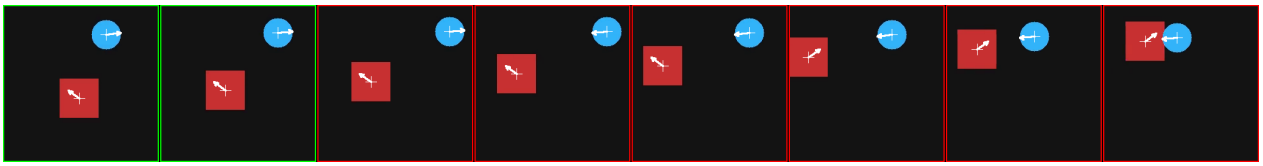


Figure 8 Bouncing Shapes dataset. Two objects move in a 2D box, with stochastic bounces at wall collisions. At each collision, the object either performs a standard bounce or reverses its velocity with probability 0.5, creating exponentially branching futures.



(a) Trajectory 1.



(b) Trajectory 2.

Figure 9 Example trajectories in the Bouncing Shapes benchmark. Green box denotes the context frame, and red box denotes the target frame for prediction.

provide examples in the Bouncing Shapes benchmark.

B.2 Metrics

Algorithm 1 Precision Error

Input: predictions $\mathcal{P} = \{p_i\}_{i=1}^{N}$, ground-truth futures $\mathcal{G} = \{g_j\}_{j=1}^{M}$, distance $\ell(\cdot, \cdot)$
Output: precision error
Initialize $\mathcal{E} \leftarrow \emptyset$. *store matched prediction errors*
for each $p \in \mathcal{P}$ **do**
  $e_p \leftarrow \min_{g \in \mathcal{G}} \ell(p, g)$. *match closest ground truth*
  $\mathcal{E} \leftarrow \mathcal{E} \cup \{e_p\}$. *add error to the set*
end for
return $\frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} e$. *average over errors*

Algorithm 2 Recall Error

Input: predictions $\mathcal{P} = \{p_i\}_{i=1}^{N}$, ground-truth futures $\mathcal{G} = \{g_j\}_{j=1}^{M}$, distance $\ell(\cdot, \cdot)$
Output: recall error
Initialize $\mathcal{E} \leftarrow \emptyset$. *store matched prediction errors*
for each $g \in \mathcal{G}$ **do**
  $e_g \leftarrow \min_{p \in \mathcal{P}} \ell(p, g)$. *match closest prediction*
  $\mathcal{E} \leftarrow \mathcal{E} \cup \{e_g\}$. *add error to the set*
end for
return $\frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} e$. *average over errors*

F1 Error. We define the F1 error as the harmonic mean of Precision Error (E_{prec}) and Recall Error (E_{rec}):

$$E_{\text{F1}} = 2 \cdot \frac{E_{\text{prec}} \cdot E_{\text{rec}}}{E_{\text{prec}} + E_{\text{rec}}}. \tag{8}$$

C FuturePerception Benchmark

Figure 10 to 14 show visual examples in our FuturePerception Benchmark for the downstream object detection task.

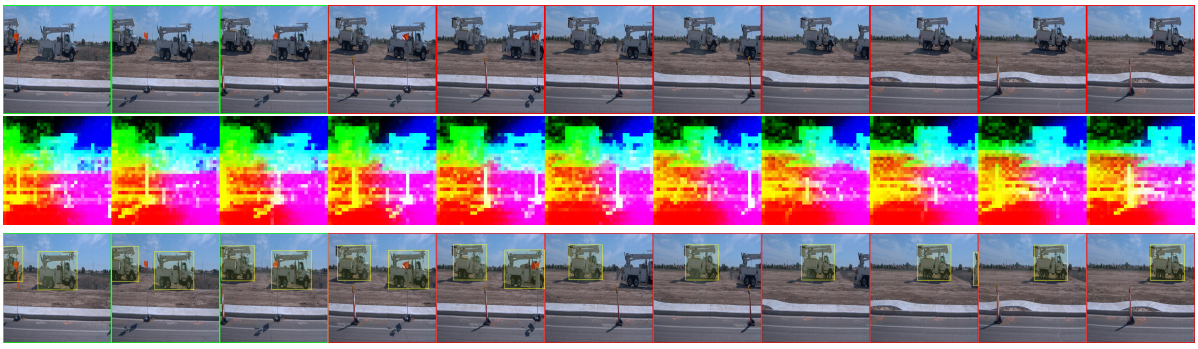


Figure 10 Waymo visual samples: original video, ground-truth latents, and oracle predictions. Green border indicates context; red border indicates future prediction.

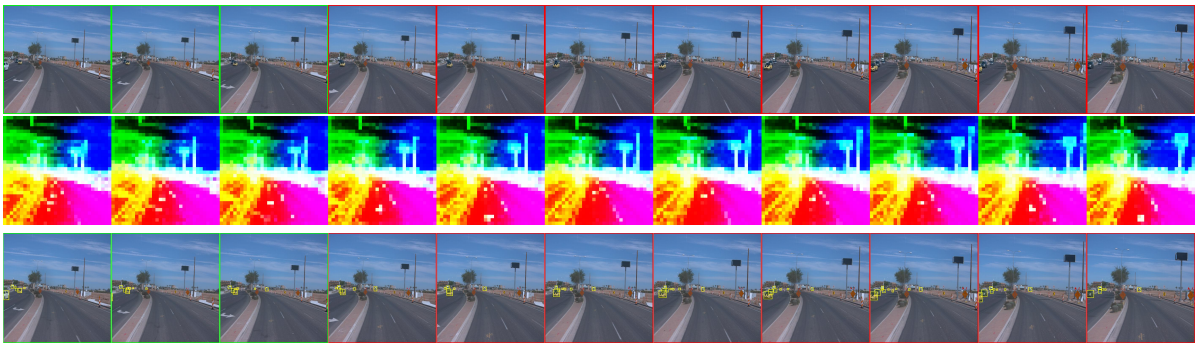


Figure 11 Waymo visual samples: original video, ground-truth latents, and oracle predictions. Green border indicates context; red border indicates future prediction.

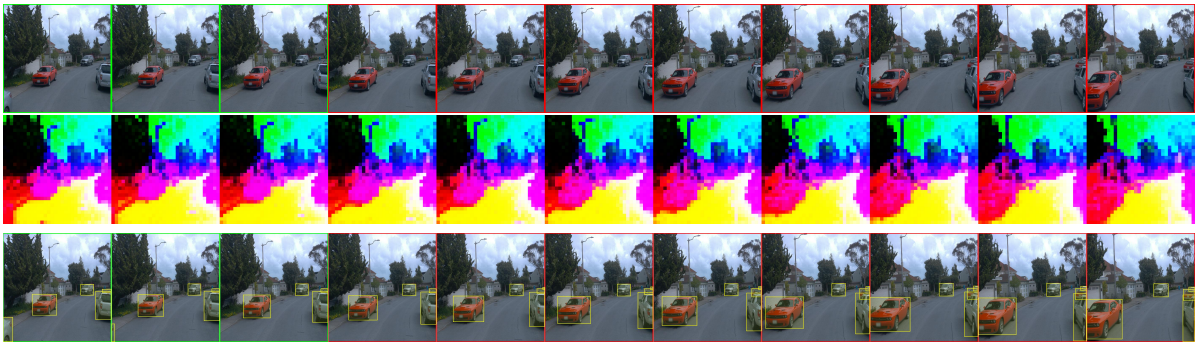


Figure 12 Waymo visual samples: original video, ground-truth latents, and oracle predictions. Green border indicates context; red border indicates future prediction.

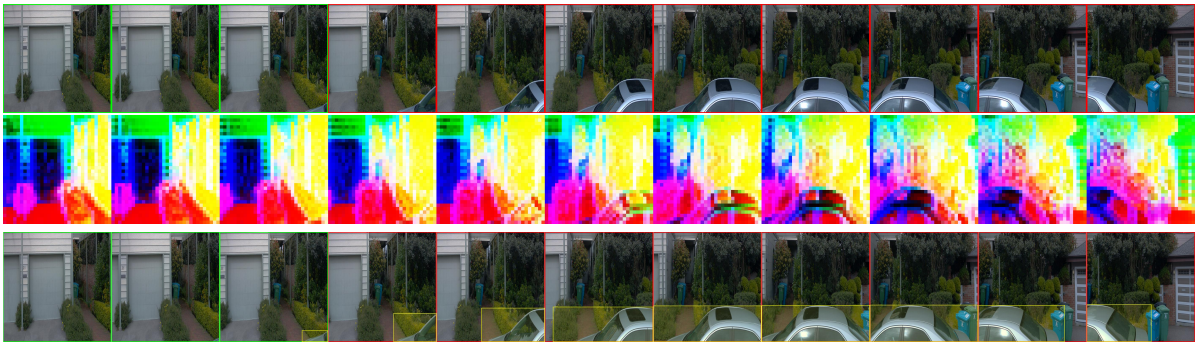


Figure 13 Waymo visual samples: original video, ground-truth latents, and oracle predictions. Green border indicates context; red border indicates future prediction.

D Detailed Ablation Results

We perform extensive ablations on FUTUREPERCEPTION object detection downstream task to verify the key design choices of our FlowWM model.

Depth and width scaling. We study the effect of model depth and width in latent world models. Varying model depth from 2 to 8 DiT layers (Figure 15a) yields no improvement in downstream detection performance, indicating that model depth is not a limiting factor for the task. In contrast, varying the width of the projection head (Figure 15b) has a pronounced effect: narrow heads severely underperform, while increasing width yields improvements up to our chosen depth $d = 1024$, beyond which gains saturate. This suggests that the width should be at least as large as the latent dimension, aligning with findings in the image domain (Zheng et al., 2025).

Temporal consistency. In Table 3, we ablate the temporal consistency loss (Eq. (5)). It improves by 5.2% relatively to the baseline, indicating that enforcing coherent temporal dynamics benefits downstream perception. Moreover, incorporating this loss via the proposed one-step projection incurs no additional computational overhead during training.

Task-driven objective. We also ablate the downstream task-driven objective, in which the detector loss is backpropagated during training on the FUTUREPERCEPTION benchmark. This provides additional performance gains, though smaller than those from the more general temporal consistency loss. We hypothesize that these gains are bounded by the limited robustness of the frozen detector to noisy predicted latents, which results in weaker gradients. Moreover, because the quality of the projected endpoint latent varies across flow-matching timesteps, appropriately weighting the task-driven loss over timesteps is important.

As shown in Table 4, increasing γ substantially improves detection performance, indicating that backpropa-

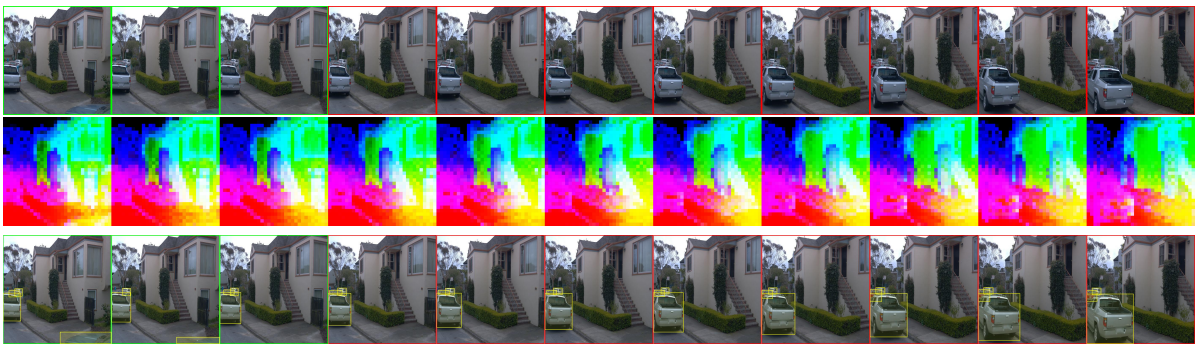


Figure 14 Waymo visual samples: original video, ground-truth latents, and oracle predictions. Green border indicates context; red border indicates future prediction.

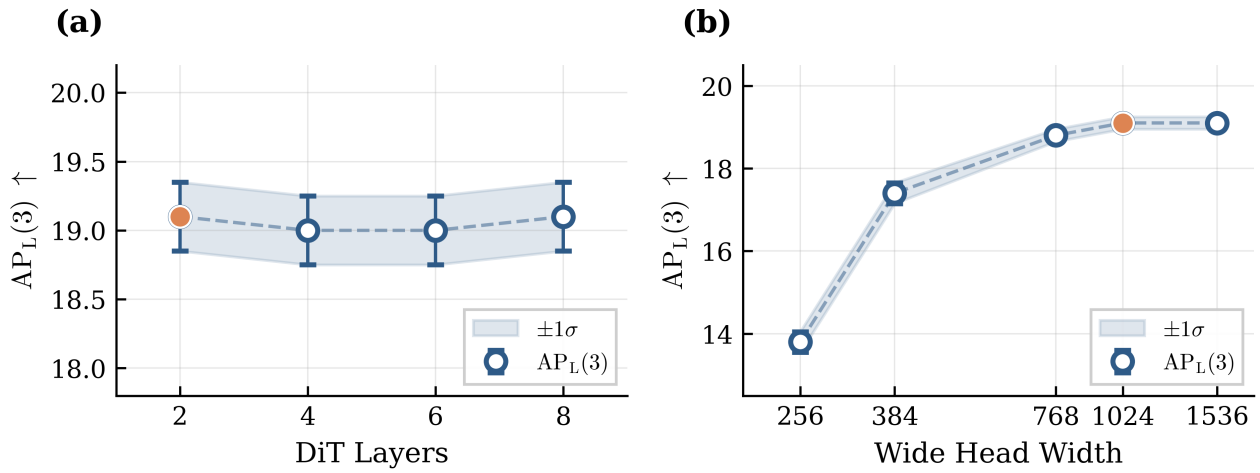


Figure 15 Ablation of model sizes on Waymo Open Dataset: depth of DiT layers (a) and width of the projection head (b). Orange indicates default configuration. Mean ± std over 3 runs.

gating the detector loss uniformly across timesteps overconstrains the Flow Matching objective. Emphasizing supervision at later timesteps allows the model to preserve a well-structured global flow while injecting task-specific constraints near the endpoint. However, excessively large values of γ lead to saturation and slight degradation in performance, suggesting a trade-off between task specialization and generative flexibility.

Shifting of timestep schedule. In Table 5, we show that skewing the training timestep distribution (Section 3.4) toward noisier timesteps is crucial. While this strategy was originally introduced to facilitate high-resolution image generation (Peebles and Xie, 2023), our results reveal that the same principle applies when increasing latent dimension and prediction horizon in video world modeling.

Scaling with number of samples. One key advantage of stochastic world models is the ability to sample multiple futures. We study how detection performance scales with the number of sampled futures N. Figure 16 shows **AP_L(N)** as a function of the sampling budget. The performance improves consistently as N increases. In contrast, deterministic world models produce the same future on the same context, and thus provide no benefit from additional samples.

E Unsuccessful Variants

We also explored several alternative training strategies and architectural variants that did not improve performance; we report them briefly for completeness.

Table 3 Ablation of training objectives on FuturePerception benchmark.

Method	$\mathbf{AP_L(3)} \uparrow$
Baseline	19.1
+ Temporal consistency loss	20.1
+ Detector loss	**20.9**

Table 4 FuturePerception: detector loss schedule ablation.

γ	$\mathbf{AP_{Large}(3)} \uparrow$
1	12.4
7	19.7
12	20.5
15	20.9
20	20.7

Reduced noise rank. We assumed that the input noise tensor of shape $[D, T, H/P, W/P]$ provides maximal stochasticity, and that the dataset's stochasticity might be lower. We therefore sampled noise only for the first frame, $[D, 1, H/P, W/P]$, and duplicated it across time. This reduced-noise variant consistently underperformed and was dropped.

ℓ_1 vs. ℓ_2 loss. We replaced the standard ℓ_2 flow-matching objective with an ℓ_1 loss. The motivation was that ℓ_2 heavily penalizes large deviations, while our predictions are often slightly off the data manifold; we hypothesized the constant-magnitude gradients of ℓ_1 might stabilize training. In practice, the change degraded performance.

F Temporal Consistency Loss

In video prediction we require *temporal coherence* across the entire sequence. The goal is not only to predict each frame correctly, but to ensure that the *predicted temporal dynamics* match those of real video latents. Formally, let the target latent sequence be

$$x_1 = (x_1^{(1)}, x_1^{(2)}, \ldots, x_1^{(T)}),$$

with $x_1^{(t)}$ the latent of frame t.
Define the temporal finite-difference operator

$$\Delta x_1^{(t)} = x_1^{(t+1)} - x_1^{(t)}.$$

A temporally consistent predictor should satisfy

$$\Delta \hat{x}_1^{(t)} \approx \Delta x_1^{(t)}, \qquad \forall\, t = 1, \ldots, T-1,$$

where $\hat{x}_1$ is the output of the flow-matching sampler.
In other words, beyond matching the final latent x_1, the model should reproduce the *temporal derivatives* (or motion patterns) implicit in the data:

$$\frac{d}{dt}\hat{x}_1(t) \approx \frac{d}{dt}x_1(t).$$

This encourages:

- consistent object motion across time,
- reduced flickering and temporal artifacts,

Table 5 Effect of timestep schedule shifting. α is the timestep skew parameter. Biasing the training distribution toward noisier timesteps substantially improves performance.

α	1	7	11	**13**	15	17
$AP_L(3)$ ↑	14.3	18.1	18.7	**19.1**	19.0	18.7

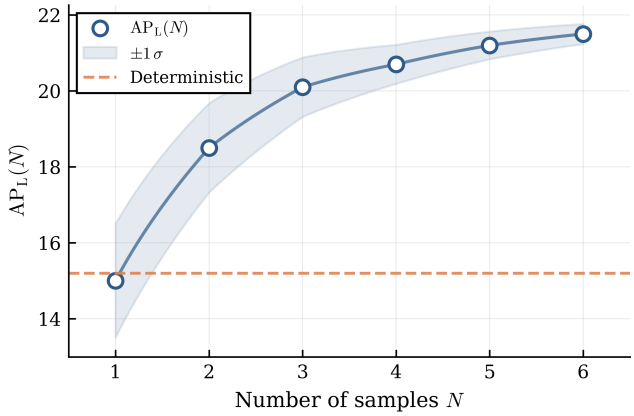


Figure 16 Sampling budget. AP_L(N) as a function of the number of sampled futures. Performance improves monotonically with sampling budget for the stochastic world model, while deterministic predictors show no gain.

- dynamics that align with the true evolution of the latent sequence.

Such temporal regularisation can be added as an auxiliary loss or incorporated directly into the velocity field parameterisation to enforce dynamical consistency.

G Task specific Loss

Time-Dependent Detector Weighting. During sampling, early steps determine coarse structure (which objects appear and where), while later steps refine higher-frequency details. Backpropagating the detector loss too early in the sampling process can overconstrain the model. A natural solution is to weight the detector loss with a factor λ_τ based on the timestep τ. We modify the training objective to:

$$\mathcal{L} = \mathcal{L}_{\text{flow}} + \lambda_\tau \, \mathcal{L}_{\text{det}},$$

and tune the schedule λ_τ.

Timestep weighting schedule for detector loss. We ablate the choice of the timestep-dependent weight λ_τ used for task-specialized training with the detector loss. We consider schedules of the form $\lambda(\tau) = \tau^\gamma$, illustrated in Figure 17. Larger values of γ place more weight on the detector loss at late timesteps (low-noise regime), where predicted latents are closer to the final semantic representation required by the detector.

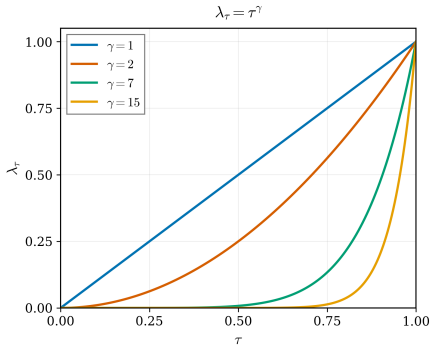


Figure 17 λ_τ for different values of γ. Larger γ increasingly emphasizes late timesteps.

H One Step Projection

Using $x_\tau = (1-\tau)x_0 + \tau x_1$, we obtain the key identity

$$\begin{aligned} x_{1,\text{pred}} - x_1 &= x_\tau - x_1 + (1-\tau)u_\theta(x_\tau, \tau) \\ &= (1-\tau)\big(u_\theta(x_\tau, \tau) - (x_1 - x_0)\big). \end{aligned} \tag{9}$$

In particular, for squared error,

$$\|x_{1,\text{pred}} - x_1\|^2 = (1-\tau)^2 \left\|u_\theta(x_\tau, \tau) - (x_1 - x_0)\right\|^2, \tag{10}$$

i.e., an endpoint loss implicitly induces a time-dependent weighting.

Auxiliary losses and time scaling. Let $R(\cdot)$ be any differentiable auxiliary loss applied at the predicted endpoint, e.g.,

$$\mathcal{L}_{\text{aux}}(\tau) = R(x_{1,\text{pred}}).$$

Using the one-step projection

$$x_{1,\text{pred}} = x_\tau + (1-\tau)\, u_\theta(x_\tau, \tau),$$

the chain rule yields a one-step parameter gradient (no unrolling):

$$\nabla_\theta \mathcal{L}_{\text{aux}} = (1-\tau)\left(\frac{\partial u_\theta(x_\tau, \tau)}{\partial \theta}\right)^\top \nabla_{x_{1,\text{pred}}} R. \tag{11}$$

Therefore, auxiliary losses applied to $x_{1,\text{pred}}$ are naturally down-weighted near $\tau \to 1$ by a factor $(1-\tau)$ (and often $(1-\tau)^2$ for losses locally proportional to the endpoint error). In practice, this motivates compensating strategies such as modifying the time-sampling distribution or explicitly reweighting $\mathcal{L}_{\text{aux}}(\tau)$ as a function of τ.

I Backpropagation through time

I.1 Backpropagation through time

Sampling from a learned flow is typically performed by numerically integrating the ODE

$$\frac{dx_\tau}{d\tau} = u_\theta(x_\tau, \tau),$$

which, under an Euler discretization with time points $\{\tau_k\}_{k=0}^N$ and $\Delta\tau_k = \tau_{k+1} - \tau_k$, reads

$$x_{k+1} = x_k + \Delta\tau_k\, u_\theta(x_k, \tau_k), \qquad k = 0, \ldots, N-1. \tag{12}$$

If an auxiliary objective $\mathcal{L}$ is applied to the final state (e.g., $\mathcal{L} = \ell(x_N)$), then the gradient $\nabla_\theta \mathcal{L}$ can be computed by backpropagating through the entire integration trajectory (BPTT). Define the adjoint variables

$$\lambda_k \coloneqq \nabla_{x_k} \mathcal{L}.$$

Because x_{k+1} depends on x_k, the chain rule through one step gives

$$\lambda_k = \left(\frac{\partial x_{k+1}}{\partial x_k} \right)^\top \lambda_{k+1}. \tag{13}$$

From (12), differentiating w.r.t. x_k yields

$$\frac{\partial x_{k+1}}{\partial x_k} = I + \Delta\tau_k \, \nabla_x u_\theta(x_k, \tau_k),$$

hence the BPTT recursion

$$\lambda_k = \Big(I + \Delta\tau_k \, \nabla_x u_\theta(x_k, \tau_k) \Big)^\top \lambda_{k+1}. \tag{14}$$

The parameter gradient accumulates contributions from every step where θ appears:

$$\nabla_\theta \mathcal{L} = \sum_{k=0}^{N-1} \Delta\tau_k \Big(\nabla_\theta u_\theta(x_k, \tau_k) \Big)^\top \lambda_{k+1}. \tag{15}$$

Practical limitations. While BPTT provides the exact gradient of the unrolled sampler, it is often difficult to use in practice: (i) it requires storing (or checkpointing) intermediate activations along the trajectory, leading to large memory overhead; (ii) the product of Jacobians in (14) can cause vanishing or exploding gradients; and (iii) it requires running the full sampler inside training. For example, using $N = 50$ integration steps makes each training iteration $\mathcal{O}(50)$ times more expensive than standard teacher-forced flow matching.

Previous work Prabhudesai et al. (2023) has explored backpropagating through a randomized number of steps K, while using LORA for finetuning less weights.

Reward Feedback Learning (ReFL) He et al. (2023) proposes a more aggressive truncation strategy for diffusion models. Specifically, ReFL samples a random diffusion timestep t, generates an on-policy latent x_t by unrolling the diffusion process from pure noise to time t without gradient tracking, and then performs a single additional denoising step with gradients enabled. This design reduces the learning signal to a single-step gradient, significantly lowering memory requirements and avoiding deep Jacobian chains. However, it still incurs the computational cost of running the diffusion sampler up to time t at each iteration, which remains expensive for large numbers of diffusion steps.

J Discussion and Limitations

Our results highlight several open challenges and directions for future work:

1. **Diffusability and structure of latent spaces.** Although we demonstrate that Flow Matching can operate directly in high-dimensional semantic latents, it is likely that some representation spaces are inherently more compatible with diffusion or flow-based modeling than others. Understanding which statistical properties make a latent space "diffusable," and how to optimize these properties, potentially through end-to-end training remains an open problem.
2. **Toward end-to-end world models.** Our current approach relies on a frozen pretrained encoder such as DINOv3. This simplifies training and avoids objective collapse, but also limits the representational quality of the predicted futures to that of the encoder. Recent works such as V-JEPA2 demonstrate that it is possible to train powerful general-purpose encoders, yet their predictors remain deterministic and difficult to scale. Unlocking stable end-to-end training of stochastic world models in high-dimensional semantic spaces represents an important challenge for the field.
3. **Action-conditioned world models and planning.** The world model developed here is conditioned on a context video. Extending it to action-conditioned settings would enable planning. With stochastic world models now tractable via Flow Matching, a key open question is how to *effectively exploit* this stochasticity:

for instance, how to use sampled futures to efficiently explore the space of possible outcomes, generate high-quality trajectory proposals, or guide downstream optimization methods such as Cross-Entropy or model-predictive control.

K Broader Impact Statement

This paper presents a methodological contribution to latent-space world modeling, focusing on stochastic generative modeling in high-dimensional pretrained semantic representations. The primary goal of this work is to advance the understanding of how uncertainty, multimodality, and semantic structure can be preserved in future prediction models evaluated through downstream perception tasks.

Potential Positive Impacts. By improving the ability of world models to represent multiple plausible futures and avoid deterministic mode collapse, this work may benefit research in areas such as robotics, autonomous systems, and model-based reinforcement learning. In particular, more faithful uncertainty and stochasticity modeling may help long horizon planning. Our benchmarks introduced in this work aim to evaluate these capacities.

Potential Risks and Misuse. The methods proposed in this paper are general-purpose modeling techniques and do not introduce new sensing, surveillance, or decision-making capabilities. However, as with many advances in predictive modeling, improved future forecasting could be incorporated into safety-critical applications such as autonomous driving or robotics. Misuse, over-reliance on model predictions, or deployment without appropriate safeguards could contribute to unsafe behavior if uncertainty is misinterpreted or ignored. Importantly, this work does not address control, policy learning, or real-time decision-making, and should not be viewed as a complete or deployable autonomous system.

Data, Privacy, and Bias Considerations. The experiments rely exclusively on existing datasets and pretrained representations. In particular, our FuturePerception is derived from the Waymo Open Dataset, which is publicly released and subject to its own data governance, licensing, and privacy protections. This work does not introduce new data collection, does not perform identity recognition, and does not aim to infer sensitive personal attributes. Any biases present in pretrained encoders or datasets may be inherited by the learned world models; mitigating such biases remains an important open research challenge beyond the scope of this paper.

Scope and Limitations. This work is intended as a research contribution rather than a deployable system. While stochastic modeling improves mode coverage and robustness over long horizons, it does not guarantee correctness or completeness of future predictions, particularly in highly uncertain or partially observed environments.

Overall, this paper advances foundational research on stochastic world modeling in semantic latent spaces. While the methods may support future applications with societal impact, their ethical implications are consistent with those commonly encountered in advancing general machine learning methodologies, and no specific risks beyond those discussed above require special mitigation at this stage.