GROUNDING GENERATED VIDEOS IN FEASIBLE PLANS VIA WORLD MODELS

Christos Ziakas*
Imperial College London

Amir Bar
UC Berkeley

Alessandra Russo
Imperial College London

ABSTRACT

Large-scale video generative models have shown emerging capabilities as zero-shot visual planners, yet video-generated plans often violate temporal consistency and physical constraints, leading to failures when mapped to executable actions. To this end, we propose Grounding Video Plans with World Models (GVP-WM), a planning method that grounds video-generated plans into feasible action sequences using a pre-trained action-conditioned world model. At test-time, GVP-WM first generates a video plan from initial and goal observations, then projects the video guidance onto the manifold of dynamically feasible latent trajectories via video-guided latent collocation. In particular, we formulate grounding as a goal-conditioned latent-space trajectory optimization problem that jointly optimizes latent states and actions under world-model dynamics, while preserving semantic alignment with the video-generated plan. Empirically, GVP-WM recovers feasible long-horizon plans from zero-shot image-to-video–generated and motion-blurred videos that violate physical constraints, across navigation and manipulation simulation tasks.

1 INTRODUCTION

Large video generative models have demonstrated strong zero-shot capabilities in synthesizing realistic and temporally coherent videos across a wide range of domains (Ho et al., 2022; Rombach et al., 2022; OpenAI, 2024; Wan Team, 2025). Beyond video synthesis, diffusion video models can generate expert-like video plans, leveraging their capabilities in prompt instruction following, object permanence, and physical consistency (Du et al., 2023a; Ko et al., 2023; Chen et al., 2025; Jang et al., 2025). Recent work further suggests that diffusion-based video models may exhibit emergent visual reasoning capabilities (Wiedemer et al., 2025). However, generated videos are often physically infeasible (e.g., object teleportation) or temporally inconsistent (e.g., motion blur), leading to violations of real-world dynamics, particularly under out-of-distribution conditions (Huang et al., 2024; Bansal et al., 2025; Agarwal et al., 2025). Despite advances in diffusion-based video models for temporal coherence (Chen et al., 2024; Song et al., 2025) and kinematics conditioning (Hu et al., 2024b; Bai et al., 2025), inferring actions directly from video-generated plans may violate real-world dynamics (Chen et al., 2025; Ni et al., 2025). Prior work leveraged video-generated plans as subgoals for model-predictive control (Nair & Finn, 2020) and hierarchical reinforcement learning (Black et al., 2023). However, these approaches assume that visual subgoals are feasible during execution, without accounting for the quality of the underlying video plans. In contrast, recent work (Luo & Du, 2025) grounds video-generated plans via goal-conditioned exploration during policy learning, enabling divergence from video guidance to ensure feasibility at the cost of additional training with environment interaction.

In this work, we introduce Grounding Video Plans with World Models (GVP-WM), a planning method that grounds video-generated plans into feasible actions at test time using pre-trained action-conditioned world models (Ha & Schmidhuber, 2018). GVP-WM projects video-generated plans onto the manifold of dynamically feasible latent trajectories under the dynamics of world models via video-guided latent collocation (Rybkin et al., 2021). Latent collocation formulates goal-conditioned planning as a trajectory optimization problem, optimizing a latent trajectory using video plans as

*Corresponding author: c.ziakas24@imperial.ac.uk

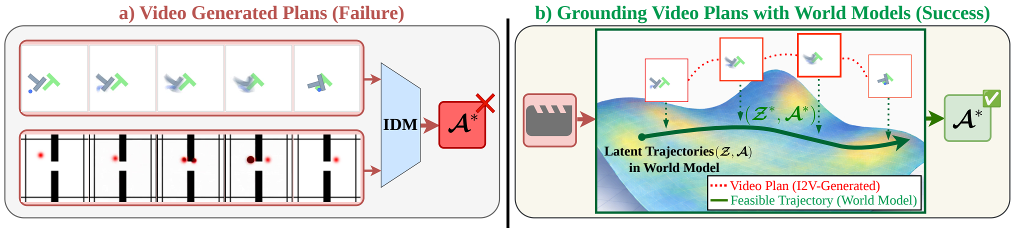


Figure 1: **(a)** Zero-shot I2V-generated plans violate temporal consistency and physical constraints, leading to failures when mapped to executable actions via inverse-dynamics models (IDM). **(b)** GVP-WM projects video-generated plans onto dynamically feasible latent trajectories via latent-space trajectory optimization under a pre-trained action-conditioned world model.

guidance, subject to world-model dynamics constraints. In fact, GVP-WM jointly optimizes both latent states and actions, as latent states inferred from video plans may violate world-model dynamics constraints. GVP-WM incorporates video guidance during latent collocation by initializing the latent trajectory from the video plan and penalizing scale-invariant semantic deviation between the optimized latent states and the video plan. Our main contributions are as follows:

- We propose GVP-WM, a test-time method for grounding video-generated plans into physically feasible action sequences using a pre-trained action-conditioned world model.
- We formulate the grounding of video plans as a latent-space trajectory optimization problem under world-model dynamics, projecting video-generated plans onto feasible latent trajectories while preserving semantic alignment with the video plan.
- We empirically demonstrate that GVP-WM outperforms video-to-action inverse-dynamics models on navigation and manipulation simulation tasks under image-to-video–generated and motion-blurred video plans, particularly in long-horizon settings.

2 Preliminaries

2.1 Goal-Conditioned Planning

We formulate planning as a finite-horizon goal-conditioned Markov Decision Process (Sutton et al., 1998; Finn & Levine, 2017), defined by the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, p, \mathcal{G}, \mathcal{C}, \mathcal{T} \rangle$. $\mathcal{S}$ denotes the state space, $\mathcal{A}$ the continuous action space, and $\mathcal{G} \subset \mathcal{S}$ the goal space. The environment dynamics are defined by an unknown stochastic transition function $p(s_{t+1} \mid s_t, a_t)$. The goal-conditioned cost function $\mathcal{C} : \mathcal{S} \times \mathcal{A} \times \mathcal{G} \to \mathbb{R}$ assigns a scalar cost for taking action a in state s under goal state g. Given an initial state s_0 and a goal $g \in \mathcal{G}$, goal-conditioned planning finds an action sequence $a_{0:T-1}$ that minimizes $\mathbb{E}\left[\sum_{t=0}^{T-1} C(s_t, a_t, g)\right]$ over a finite horizon T.

2.2 Latent Collocation in World Models

Model-based planning approximates environment dynamics using an action-conditioned world model in latent space, as planning directly in pixel space is computationally infeasible (Hafner et al., 2020; LeCun, 2022). In particular, an action-conditioned latent world model consists of an encoder $E_\phi : \mathcal{S} \to \mathcal{Z}$ that maps observations to compact latent states $z_t = E_\phi(s_t)$, and a transition function f_ψ that predicts future latent states conditioned on a history of H latent states and actions $z_{t+1} = f_\psi(z_{t-H:t}, a_{t-H:t})$. Planning in latent world models minimizes a latent cost objective $\tilde{C} : \mathcal{Z} \times \mathcal{A} \times \mathcal{Z} \to \mathbb{R}$, defined over latent trajectories, measuring progress from the initial state s_0 to the goal state g in latent space using E_ϕ. In contrast to shooting-based or gradient-based planning methods, where intermediate latent states are implicitly determined by forward simulation of actions under the world model, latent collocation treats both latent states and actions as explicit decision variables (Rybkin et al., 2021). In particular, latent collocation formulates planning as a latent-space trajectory optimization problem that jointly optimizes latent (knot) states $z_{0:T}$ and actions $a_{0:T-1}$ to minimize $\tilde{C}$ subject to learned world-model dynamics constraints.

3 Grounding Video Plans with World Models

Our method first generates a video plan from the initial and goal observations and then grounds it into a feasible action sequence using trajectory optimization in latent space. We provide pseudocode for GVP-WM in Algorithm 2, with a detailed version included in the appendix B

3.1 Problem Formulation

Video models may produce video plans with temporally inconsistent or blurry transitions, particularly in out-of-distribution settings (Huang et al., 2024; Bansal et al., 2025). We consider the problem of grounding a video plan τ_{vid} into a feasible trajectory of latent states $z_{0:T}$ and actions $a_{0:T-1}$ using an action-conditioned world model (E_ϕ, f_ψ). We formulate this task as a finite-horizon goal-conditioned MDP, defined in Section 2.1, where the state space $\mathcal{S}$ consists of high-dimensional visual observations o_t and, optionally, low-dimensional proprioceptive information p_t. The goal space $\mathcal{G}$ is defined over visual observations, where each goal $g \in \mathcal{G}$ corresponds to a target image o_g. Given an initial state s_0, a goal observation o_g, and a video-generated plan τ_{vid}, our objective is to optimize a latent trajectory that satisfies the world model dynamics f_ψ, while following the semantic guidance provided by the video plan τ_{vid}. In this paper, we formulate this grounding process as a constrained trajectory optimization problem in latent space, using video-guided latent collocation as described below.

3.2 Video-Guided Latent Collocation

Our goal is to find a latent trajectory of $z_{0:T}$ and $a_{0:T-1}$ that is guided by a video plan τ_{vid} while satisfying the environment transition dynamics of the world model f_ψ.

3.2.1 Video Plan Generation

A video plan τ_{vid} is a sequence of visual observations generated by a conditional video generative model $\mathcal{G}$ that provides temporally ordered visual foresight for completing a task. $\mathcal{G}$ can be instantiated using an image-to-video (I2V) diffusion-based video model that produces temporally coherent visual transitions between the start and goal observations (Blattmann et al., 2023). In particular, I2V diffusion models with first–last frame conditioning employ a masking mechanism that fixes the initial and final frames as the start and goal observations, guiding the generation of intermediate frames while maintaining temporal consistency (Wan Team, 2025). Given an initial visual observation o_0 and a visual goal observation o_g, we use a diffusion-based I2V generation model $\mathcal{G}$ trained with first–last frame transformation to synthesize a video plan τ_{vid} as

$$\tau_{\text{vid}} = \{o_0, \tilde{o}_1, \ldots, \tilde{o}_{T-1}, o_g\} \sim \mathcal{G}(\cdot \mid o_0, o_g, c), \tag{1}$$

where c denotes optional contextual information, such as a textual description of the task. The video model can optionally be fine-tuned on a dataset of domain-specific demonstrations $\mathcal{D}_{\text{demo}}$ to improve physical consistency. If the video generation model and the world model operate at different frame-skip ratios, temporal subsampling is applied to the video plan to ensure temporal alignment.

3.2.2 Video Guidance

We map the generated video plan into the latent state space of the action-conditioned world model using its underlying visual encoder, yielding a sequence of latent states $z^{\text{vid}}_{0:t}$. However, the latent representations derived from generated video frames may exhibit magnitude drift relative to the in-distribution latent states of the pre-trained world model. To improve robustness under distribution shift, we opt for world models with pre-trained frozen visual encoders (Zhou et al., 2025). In addition, we introduce a scale-invariant semantic alignment loss, which is equivalent to cosine similarity. Specifically, we project both optimized and video latent states onto the unit ℓ_2 hypersphere by applying $\phi(z) = z/\|z\|_2$ as follows:

$$\mathcal{L}_{\text{vid}}(z_t, z_t^{\text{vid}}) = \left\|\phi(z_t) - \phi(z_t^{\text{vid}})\right\|^2 = 2\big(1 - \cos\theta_t\big), \tag{2}$$

Algorithm 1 GVP-WM

Require: s_0; o_g; $\mathcal{G}$; (E_ϕ, f_ψ), (T, K)

Generate video plan: $\tau_{\text{vid}} \sim \mathcal{G}(\cdot \mid o_0, o_g, c)$
Encode video: $z^{\text{vid}}_{0:T} \leftarrow E_\phi(\tau_{\text{vid}})$
Init primals: $\mathcal{Z} \leftarrow z^{\text{vid}}_{0:T}$, $\mathcal{A} \leftarrow \mathbf{0}$
for $t = 0$ **to** $T-1$ **step** K **do**
    Current latent state: $z_t \leftarrow E_\phi(s_t)$
    Init duals: $\lambda \leftarrow \mathbf{0}$, $\rho \leftarrow \rho_0$
    for $k = 1$ **to** O_{ALM} **do**
        for $i = 1$ **to** I_{ALM} **do**
            Update $(\mathcal{Z}, \mathcal{A})$ via $\nabla \mathcal{L}_\rho$ (Eq. 4)
        end for
        Update Duals: $\lambda \leftarrow \lambda + \rho \mathcal{L}_{\text{dyn}}$
        Update Penalty: $\rho \leftarrow \min(\gamma\rho, \rho_{\max})$
    end for
    Execute $a_{t:t+K}$ from $\mathcal{A}$ and set state s_{t+K}
end for

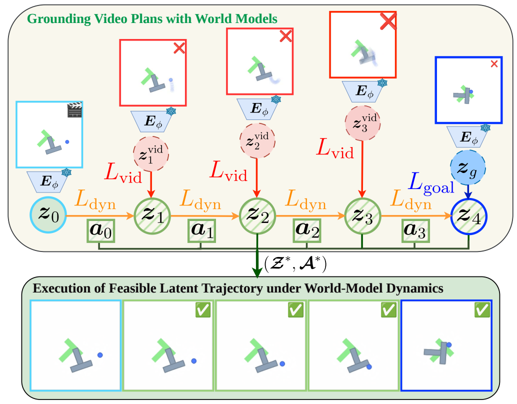


Figure 2: Overview of GVP-WM. Video-guided latent collocation optimizes both latent states $z_{t+1:T}$ and actions $a_{t:T-1}$ by minimizing Eq. 4, balancing video alignment (L_{vid}) and world-model dynamics (L_{dyn}).

where θ_t denotes the angle between z_t and z_t^{vid}. This objective penalizes angular deviation (Grill et al., 2020), while remaining scale-invariant to latent magnitude.

3.2.3 Grounding via Latent Collocation

We formulate grounding video plans into feasible action sequences as a video-guided direct collocation problem. In direct collocation, both the latent states $\mathcal{Z} = z_0, \dots, z_T$ and the actions $\mathcal{A} = a_0, \dots, a_{T-1}$ are treated as decision variables, as described in Section 2.2. Our goal is to solve for a trajectory $(\mathcal{Z}^*, \mathcal{A}^*)$ that minimizes the divergence between the latent states of the optimized trajectory and the video plan z^{vid}, while satisfying the world model dynamics $f\psi$. To this end, we instantiate the latent cost objective $\tilde{C}$ defined in 2.2 as a weighted combination of a video alignment loss, a terminal goal loss, and an action regularization term, leading to the following constrained optimization problem:

$$\begin{aligned} \min_{\mathcal{Z},\mathcal{A}} \quad & \lambda_{\text{v}} \sum_{t=1}^{T-1} \mathcal{L}_{\text{vid}}(z_t, z_t^{\text{vid}}) + \lambda_{\text{g}} \mathcal{L}_{\text{goal}}(z_T, z_g) + \lambda_{\text{r}} \sum_{t=0}^{T-1} \|a_t\|^2 \\ \text{s.t.} \quad & z_{t+1} = f_\psi(z_{t-H:t}, a_{t-H:t}), \quad \forall t \in \{0, \dots, T-1\}, \\ & a_{\min} \preceq a_t \preceq a_{\max}, \quad \forall t \in \{0, \dots, T-1\}. \end{aligned} \tag{3}$$

In our formulation, the dynamics of the latent action-conditioned world model f_ψ are enforced as hard constraints. The video alignment loss $\mathcal{L}_{\text{vid}}$ corresponds to the objective defined in Eq. 2. The goal loss $\mathcal{L}_{\text{goal}}$ is defined as the mean squared error between the terminal latent state z_T and the latent representation of a visual goal z_g. The coefficients λ_v, λ_g, and λ_r are hyperparameters that balance the contribution of the loss terms. We empirically validate the impact of video guidance, latent collocation, and scale-invariant semantic alignment via ablation studies in Appendix D.

3.3 Visual Model Predictive Control

At each timestep t, we solve the video-guided latent collocation optimization problem over the planning horizon, producing the optimal trajectory $(z^*_{t+1:T}, a^*_{t:T-1})$. We then execute actions with Model Predictive Control (MPC).

3.3.1 OPTIMIZATION VIA AUGMENTED LAGRANGIAN

To efficiently solve the non-linear constrained optimization problem described in Equation 3, we employ the Augmented Lagrangian Method (ALM) (Bertsekas, 1982). The loss is defined as:

$$\mathcal{L}_\rho(\mathcal{Z}, \mathcal{A}, \Lambda) = \tilde{C}(\mathcal{Z}, \mathcal{A}) + \sum_{t=0}^{T-1} \left(\lambda_t^\top \mathcal{L}_{\text{dyn}}^t + \frac{\rho}{2} \|\mathcal{L}_{\text{dyn}}^t\|^2 \right), \tag{4}$$

where $\tilde{C}$ denotes the latent cost objective from Eq. 3. $\Lambda = \{\lambda_0, \ldots, \lambda_{T-1}\}$ are the Lagrange multipliers, and $\rho > 0$ is a scalar penalty parameter. The term $\mathcal{L}_{\text{dyn}}^t$ corresponds to the dynamics constraint violation at timestep t, defined as:

$$\mathcal{L}_{\text{dyn}}^t(\mathcal{Z}, \mathcal{A}; f_\psi) = z_{t+1} - f_\psi(z_{t-H:t}, a_{t-H:t}). \tag{5}$$

Optimization proceeds using a standard primal–dual approach (Bertsekas, 1982), alternating between gradient-based updates of the primal variables $(\mathcal{Z}, \mathcal{A})$ and the dual variables Λ. In particular, we perform I_{ALM} inner (primal) iterations per outer (dual) step and run the optimization for O_{ALM} outer iterations using Adam (Kingma & Ba, 2015). We employ a geometric continuation schedule for ρ to progressively enforce dynamic constraints, increasing the penalty at each outer iteration with penalty growth rate $\gamma > 1$ up to a maximum value $\rho_{\max}$. This schedule enables prioritizing video guidance in early iterations, while progressively enforcing physical consistency as ρ increases. For the bounded action constraints, we apply a smooth action reparameterization, following prior work in continuous-control optimization (Haarnoja et al., 2018). We empirically validate this smooth reparameterization against projected gradient descent in the ablation study in Appendix D.

3.3.2 EXECUTION VIA MODEL PREDICTIVE CONTROL

The ALM optimization solves for a dynamically feasible latent state–action trajectory $(\mathcal{Z}, \mathcal{A})$ that realizes the latent states of the video plan z_{vid}. The latent state variables $\mathcal{Z}$ are initialized from the latent states of the video plan z_{vid}. This initialization anchors the optimization to a semantic prior, improving the recovery of feasible plans, as shown in Ablation 4.5. When low-level proprioceptive information is available, we initialize the corresponding states using a static prior at the current state s_t. Upon convergence, the optimization yields an optimized latent trajectory and action sequence $(\mathcal{Z}^*, \mathcal{A}^*)$, where the action sequence $\mathcal{A}^*$ is executed using a receding-horizon MPC. Conditioning video-guided latent collocation on the current state reduces error accumulation over long horizons, since the current state is enforced through the dynamics constraints in latent-space trajectory optimization. Optionally, we refine the optimized action sequence $\mathcal{A}^*$ by sampling C candidate sequences from $\mathcal{N}(\mathcal{A}^*, \sigma^2 I)$, evaluating them via open-loop rollouts under the latent dynamics f_ψ, and selecting the sequence with the lowest terminal goal cost. Given an execution horizon K, GVP-WM executes the first K actions from the optimal trajectory. We analyze the impact of MPC and local refinement via ablations in Appendix D.

The ALM optimization solves for a dynamically feasible latent state–action trajectory $(\mathcal{Z}, \mathcal{A})$ that realizes the latent states of the video plan z_{vid}. The latent state variables $\mathcal{Z}$ are initialized from the latent states of the video plan z_{vid}. This initialization anchors the optimization to a semantic prior, improving the recovery of feasible plans, as shown in Ablation 4.5. When low-level proprioceptive information is available, we initialize the corresponding states using a static prior at the current state s_t. Upon convergence, the optimization yields an optimized latent trajectory and action sequence $(\mathcal{Z}^*, \mathcal{A}^*)$, where the action sequence $\mathcal{A}^*$ is executed using a receding-horizon MPC. Conditioning video-guided latent collocation on the current state reduces error accumulation over long horizons, since the current state is enforced through the dynamics constraints in latent-space trajectory optimization. Optionally, we refine the optimized action sequence $\mathcal{A}^*$ by sampling C candidate sequences from $\mathcal{N}(\mathcal{A}^*, \sigma^2 I)$, evaluating them via open-loop rollouts under the latent dynamics f_ψ, and selecting the sequence with the lowest terminal goal cost. Given an execution horizon K, GVP-WM executes the first K actions from the optimal trajectory. We analyze the impact of MPC and local refinement via ablations in Appendix D.

4 EXPERIMENTS

We evaluate GVP-WM along three axes: (i) its ability to ground zero-shot video plans into feasible actions under out-of-distribution environments, (ii) robustness to temporally inconsistent video guidance, and (iii) performance in long-horizon settings. We compare against both world-model-based planners without video guidance and direct video-to-action baselines. We further provide ablation studies analyzing the contributions of video guidance and the latent collocation formulation, as well as a qualitative evaluation.

4.1 EXPERIMENTAL SETUP

4.1.1 ENVIRONMENTS AND EVALUATION SETUP

We evaluate GVP-WM in two control environments across multiple long-horizon settings, following the evaluation protocols established in prior work (Zhou et al., 2025). Push-T Chi et al. (2025) is a contact-rich 2D manipulation task in which an agent pushes a T-shaped object to a target configuration, while Wall is a 2D navigation task that requires visual planning. For both environments, we use DINO-WM (Zhou et al., 2025), a pre-trained action-conditioned world model that leverages DINOv2 (Oquab et al., 2023) as a frozen visual encoder. Video plans are generated using Wan2.1-FLF2V-14B (720p) image-to-video diffusion model Wan Team (2025) in a zero-shot setting (WAN-0S). Since both environments are out-of-distribution relative to the internet-scale pretraining data of the video model, we additionally fine-tune the video model using $\mathcal{D}_{\text{demo}} = 100$ task-specific demonstrations via LoRA (Hu et al., 2022), and generate video plans with domain-adapted video guidance (WAN-FT). In addition, we consider video plans obtained from expert policies that are in-distribution with respect to world-model training (ORACLE). To evaluate robustness to temporally inconsistent video guidance, we introduce synthetic motion blur by temporally averaging k consecutive frames (MB-k). In contrast to prior work (Zhou et al., 2025), goal states are specified as visual observations, with no access to proprioceptive goal information during planning. We evaluate planning performance for horizons $T \in 25, 50, 80$ in Push-T and $T \in 25, 50$ in Wall, using receding-horizon execution. For each setting, we evaluate on 50 initial–goal pairs sampled from expert policy trajectories of the corresponding horizon length. We report Success Rate, defined as the fraction of successful rollouts across all evaluation episodes. Full implementation details are provided in Appendix E.

Table 1: SR for planning horizons T. WAN-0S: zero-shot; WAN-FT: domain-adapted; ORACLE: expert (upper bound).

METHOD	PUSHT T=25	PUSHT T=50	PUSHT T=80	WALL T=25	WALL T=50
MPC-CEM	<u>0.74</u>	<u>0.28</u>	**0.06**	<u>0.92</u>	0.74
MPC-GD	0.32	0.04	0.00	0.04	0.04
UNIPI (WAN-0S)	0.00	0.00	0.00	0.30	0.14
UNIPI (WAN-FT)	0.10	0.00	0.00	0.40	0.18
UNIPI (*ORACLE*)	*0.52*	*0.18*	*0.08*	*1.00*	*1.00*
GVP-WM (WAN-0S)	0.56	0.12	<u>0.04</u>	0.86	<u>0.76</u>
GVP-WM (WAN-FT)	**0.80**	**0.30**	**0.06**	**0.94**	**0.90**
GVP-WM (*ORACLE*)	*0.98*	*0.72*	0.36	*1.00*	*1.00*

Table 2: SR under increasing levels of motion blur. MB-k denotes temporal averaging over k consecutive frames.

SOURCE	METHOD	PUSHT T=25	PUSHT T=50	PUSHT T=80	WALL T=25	WALL T=50
MB-10	UNIPI	0.03	0.00	0.00	0.00	0.02
	GVP-WM	**0.82**	**0.46**	**0.08**	**0.94**	**1.00**
MB-5	UNIPI	0.04	0.00	0.06	0.40	0.52
	GVP-WM	**0.94**	**0.54**	**0.16**	**1.00**	**1.00**
MB-3	UNIPI	0.16	0.00	0.00	0.52	0.82
	GVP-WM	**0.96**	**0.70**	**0.34**	**1.00**	**1.00**
ORACLE	UNIPI	0.52	0.18	0.08	**1.00**	**1.00**
	GVP-WM	**0.98**	**0.72**	**0.36**	**1.00**	**1.00**

4.1.2 BASELINES

We compare against two MPC-based planners that use the same pre-trained action-conditioned world model (DINO-WM), without relying on video guidance. MPC-CEM is a shooting-based method that uses the Cross-Entropy Method to optimize action sequences by sampling candidate trajectories and evaluating them via open-loop rollouts in the latent world model. MPC-GD is a gradient-based planner that optimizes action sequences by backpropagating through the differentiable world-model dynamics. For both methods, planning is performed in the learned latent world model with receding-horizon execution ($K = 1$). We also compare against UniPi (Du et al., 2023a), a direct video-to-action baseline based on inverse dynamics that maps video trajectories to actions without enforcing world-model dynamics. For a fair comparison, UniPi is provided with the same video plans as GVP-WM; thus, the methods differ only in whether video guidance is executed directly or grounded through a learned world model. Further details are provided in Appendix E.

4.2 Grounding I2V-Generated Video Plans

Table 1 reports success rates on Push-T and Wall across increasing planning horizons for all methods and sources of video guidance. We compare GVP-WM against MPC-based planners that do not use video guidance (MPC-CEM, MPC-GD), as well as a direct video-to-action baseline based on inverse dynamics (UniPi). For methods that use video guidance (GVP-WM and UniPi), we consider three sources of video plans: zero-shot generated videos (WAN-0S), domain-adapted generated videos (WAN-FT), and in-distribution expert videos (ORACLE) serving as upper bound. Under domain-adapted video guidance, GVP-WM outperforms MPC-CEM in all evaluated settings except Push-T at $T{=}80$, where performance is comparable. In the zero-shot setting, video guidance is out-of-distribution for the video model in both environments. Consequently, for WAN-0S, MPC-CEM performs better overall, except on Wall at $T{=}50$. However, GVP-WM requires substantially less planning time than MPC-CEM. Under both zero-shot and domain-adapted video guidance, GVP-WM consistently outperforms MPC-GD across environments and planning horizons, particularly on Wall, where gradient-based planning exhibits limitations in visual navigation.

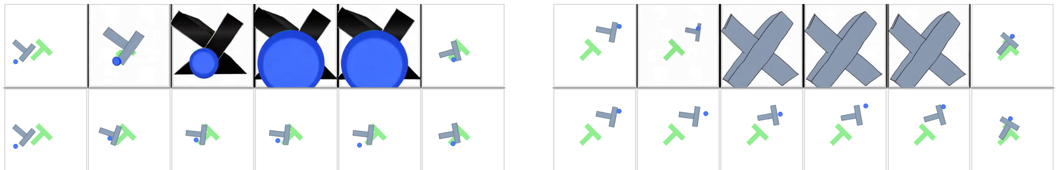
GVP-WM outperforms UniPi across both tasks and planning horizons. For Push-T, UniPi fails under zero-shot and domain-adapted video guidance. In contrast, GVP-WM with domain-adapted video guidance achieves strong performance at horizons $T{=}25$ and 50. Under zero-shot guidance, GVP-WM performance decreases but still recovers feasible plans, indicating robustness. We observe similar trends on Wall under video-generated guidance, with higher performance than Push-T. With in-distribution expert videos, GVP-WM outperforms UniPi on Push-T at all horizons. On Wall, UniPi and GVP-WM achieve perfect success provided with oracle videos at both horizons, indicating that inferring actions from video plans is more effective for tasks relying on visual planning than for contact-rich manipulation. In addition, UniPi has negligible inference time, requiring a single forward pass. We conclude that GVP-WM, unlike UniPi, recovers feasible actions from image-to-video–generated plans across both environments by grounding video plans in world-model dynamics.

4.3 Robustness to Motion Blur

To evaluate robustness to temporally inconsistent video guidance, we introduce synthetic motion blur into expert videos by temporally averaging consecutive frames over sliding windows of size 3, 5, and 10 (MB-3, MB-5, MB-10). Table 2 reports success rates for UniPi and GVP-WM under increasing levels of temporal blur. UniPi is highly sensitive to temporally degraded guidance. At horizon $T{=}25$, success drops from 0.52 under clean expert videos to 0.16, 0.04, and 0.02 under MB-3, MB-5, and MB-10, respectively. At $T{=}50$, UniPi achieves only 0.18 success under clean videos and fails entirely under any level of blur. In contrast, GVP-WM remains robust to temporal inconsistencies in video guidance. At $T{=}25$, it maintains high success under mild and moderate blur (0.96 and 0.94) and retains strong performance even under severe blur (0.82). At $T{=}50$, performance degrades gradually as blur increases, from 0.72 under clean videos to 0.70, 0.54, and 0.46 under MB-3, MB-5, and MB-10. Across all settings, GVP-WM substantially outperforms UniPi, with the performance gap widening at longer horizons and under stronger temporal blur. These results show that GVP-WM remains effective when video guidance exhibits temporal inconsistencies that violate physical constraints, whereas inverse-dynamics models fail under such conditions.

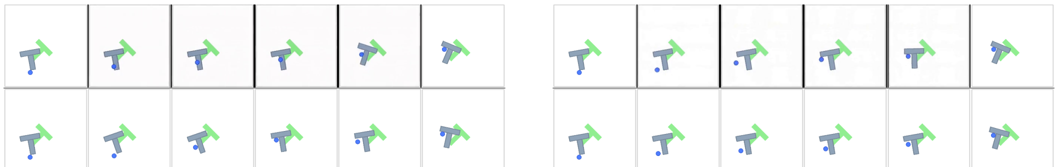
4.4 Qualitative Analysis of I2V-generated Video Plans

Figure 3 compares image-to-video (I2V) plans generated by the Wan2.1-FLF2V-14B diffusion model with execution trajectories produced by GVP-WM. In our experimental setup, zero-shot I2V-generated plans frequently violate physical constraints, as both Push-T and Wall simulation environments are out of distribution relative to the internet-scale pretraining data of the video model. In Push-T, zero-shot plans often exhibit morphological drift, causing the appearance or identity of the manipulated object to change across frames. GVP-WM can sometimes recover a feasible trajectory by enforcing world-model dynamics and disregarding inconsistent video frames (Figure 3a); however, severe semantic drift can still lead to execution failure (Figure 3b). In addition, zero-shot video plans



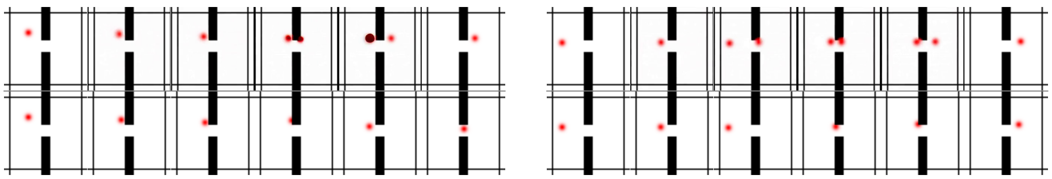
(a) Morphological Drift (Success)

(b) Morphological Drift (Failure)



(c) Rigid-Object Physics Violation (Failure)

(d) Domain-Adapted Video Guidance (Success)



(e) Spatial Bilocation (Failure)

(f) Spatial Bilocation (Success)

Figure 3: Qualitative comparison of video-generated plans (top) and GVP-WM executions (bottom). In Push-T, zero-shot video plans (WAN-0S) exhibit physical inconsistencies, including morphological drift (a–b) and rigid-object physics violations (c), while domain-adapted video guidance (WAN-FT) produces physically consistent plans. In Wall, zero-shot video plans exhibit spatial bilocation (e-f).

can violate rigid-object physics (Figure 3c). In contrast, domain-adapted video guidance from a diffusion model fine-tuned with LoRA on a small dataset (Section 4.1.1) produces more physically consistent plans, which GVP-WM is able to follow to successfully complete the task (Figure 3d). In Wall, zero-shot video plans can additionally exhibit spatial bilocation, where the agent appears at multiple spatial locations within the same frame; despite this violation, GVP-WM is still able to recover feasible execution trajectories in some cases (Figure 3f). Additional qualitative examples and failure modes across both environments are provided in Appendix A.

4.5 Ablation Studies

We study the impact of video guidance and latent collocation, with results and additional ablations on GVP-WM design choices reported in Appendix D. Performance depends on video quality, while latent state initialization and joint optimization of latent states and actions are critical for feasibility. Removing latent state initialization substantially degrades performance under both domain-adapted and oracle guidance. Fixing latent states to video plans and optimizing only actions leads to performance collapse, indicating that video-generated latent trajectories are not dynamically feasible and cannot serve as waypoints, consistent with inverse-dynamics failures (Table 1).

5 Related Work

5.1 Video Models for Decision Making

Large-scale generative video models (Ho et al., 2022; Blattmann et al., 2023) have been widely explored as zero-shot visual planners. LVP (Chen et al., 2025) proposes a foundation video model for robotics, leveraging diffusion history guidance (Song et al., 2025) and diffusion forcing (Chen et al., 2024) to improve temporal coherence. Other approaches enhance video generation by conditioning on pose (Bai et al., 2025), motion (Hu et al., 2024b), or physical constraints (Yuan et al., 2023). Beyond video generation, several works infer actions directly from generated videos (Baker et al., 2022; Du et al., 2023a; Ko et al., 2023). UniPi (Du et al., 2023a) synthesizes expert demonstrations via text-conditioned video generation and maps video frames to actions using an inverse dynamics model. However, despite progress in inferring actions from videos (Edwards et al., 2019; Pavlakos et al., 2024; Potamias et al., 2025; Li et al., 2025; Zhang et al., 2025), mapping video-generated plans to feasible action sequences remains a challenge (Chen et al., 2025; Ni et al., 2025). In contrast, hierarchical approaches treat generated videos as high-level guidance rather than direct action supervision. HVF (Nair & Finn, 2020) generates visual subgoals via video prediction for model-predictive control, while VLP (Du et al., 2023b) performs tree search over generated video futures to solve long-horizon tasks. SuSIE (Black et al., 2023) leverages image-editing models to synthesize sparse visual subgoals that guide a low-level, goal-conditioned policy. However, these approaches assume that generated videos or visual subgoals are physically feasible in the target environment. Recent work (Luo & Du, 2025) addresses this limitation by using generated videos as training-time guidance in policy learning. In contrast, GVLP-WM grounds generated video plans at inference time by optimizing latent states and actions under a learned, action-conditioned world model, enabling test-time grounding without additional environment interaction or policy training.

5.2 Planning with World Models

World models have been widely used for learning environment dynamics across domains such as robotics (Yang et al., 2023; Agarwal et al., 2025), autonomous driving (Hu et al., 2024a; Russell et al., 2025), and video games (Bruce et al., 2024; Hafner et al., 2025). These models learn latent representations of environment dynamics conditioned on actions, supporting planning (LeCun, 2022; Hafner et al., 2019) and model-based reinforcement learning through rollouts in imagination (Hafner et al., 2023; 2025). Despite progress in goal-conditioned universal value function learning (Ma et al., 2024; Ziakas & Russo, 2026), policy learning in world models may fail to generalize due to reliance on zero-shot reward models. In contrast, planning enables inference-time optimization of action sequences via simulated rollouts under world-model dynamics, without requiring policy learning or reward functions (Hafner et al., 2020; LeCun, 2022). Shooting-based methods perform trajectory optimization by evaluating candidate action sequences (Zhou et al., 2025; Bar et al., 2025), while gradient-based methods directly optimize trajectories through the learned dynamics (Terver et al., 2026; Zhou et al., 2025). Latent collocation has been proposed as a formulation that optimizes over latent state trajectories while enforcing dynamics consistency (Rybkin et al., 2021). In this work, we propose video-guided collocation, which grounds visual plans in a learned world model and improves performance when sufficient video guidance is available. By anchoring latent trajectories to a semantic prior, GVP-WM also reduces the effective search space, leading to faster convergence.

6 Discussion, Limitation, and Future Work

GVP-WM grounds video-generated plans into physically feasible action sequences using action-conditioned world models. Large video models, while powerful, frequently violate low-level physical constraints when applied to out-of-distribution robotic environments such as Push-T, producing artifacts including object teleportation, rigid-body violations, and temporal inconsistencies. In such cases, directly mapping video frames to actions via inverse-dynamics models fails because the underlying visual plans are not dynamically feasible. Our empirical results show that GVP-WM can recover executable action sequences even when video plans are physically invalid or temporally inconsistent, and that it consistently outperforms inverse-dynamics baselines across environments and planning horizons. Performance under domain-adapted video guidance and in-distribution oracle video plans further indicates the potential of video-guided planning. These results suggest that as large video models and action-conditioned world models continue to improve, grounding video plans with world models can provide a robust mechanism for mapping video plans into executable actions.

GVP-WM relies on learned world-model dynamics; in zero-shot settings, mismatches between the learned dynamics and the environment can lead to infeasible plans regardless of video quality. Although faster than sampling-based planners, GVP-WM still requires iterative test-time optimization, which may limit applicability in real-world settings. While GVP-WM is robust to motion blur, severely misaligned zero-shot video guidance can degrade performance relative to unguided planning. Future work includes extending GVP-WM to real-world robotic manipulation, where zero-shot video guidance is in-distribution, as well as exploring hierarchical planning and policy distillation.

References

Niket Agarwal et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Yutong Bai, Danny Tran, Amir Bar, Yann LeCun, Trevor Darrell, and Jitendra Malik. Whole-body conditioned egocentric video prediction. *arXiv preprint arXiv:2506.21552*, 2025.

Bowen Baker, Ilge Akkaya, Peter Zhokhov, et al. Video pretraining (vpt): Learning to act by watching unlabeled online videos. In *NeurIPS*, 2022.

Hritik Bansal, Zongyu Lin, Tianyi Zhang, Siddharth Goyal, et al. Videophy: Evaluating physical commonsense for video generation. In *International Conference on Learning Representations (ICLR)*, 2025.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15791–15801, 2025.

Dimitri P Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 1982.

Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, et al. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Jake Bruce, Michael Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, et al. Genie: Generative interactive environments. In *International Conference on Machine Learning (ICML)*, 2024.

Boyuan Chen, Zhuo Saxena, Kushal Li, et al. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *arXiv preprint arXiv:2407.01392*, 2024.

Boyuan Chen, Peihao Li, Tianyuan Zhang, Haoran Geng, Jitendra Malik, Kiwhan Song, Caiyi Zhang, Pieter Abbeel, Russ Tedrake, et al. Large video planner enables generalizable robot control. *arXiv preprint arXiv:2512.15840*, 2025.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Yilun Du, Mengjiao Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B. Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023a.

Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, et al. Video language planning. *arXiv preprint arXiv:2310.10625*, 2023b.

Ashley Edwards, Himanshu Sahni, Yannick Schroecker, and Charles Isbell. Imitating latent policies from observation. In *International conference on machine learning*, pp. 1755–1763. PMLR, 2019.

Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 2786–2793. IEEE, 2017.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

David Ha and Jürgen Schmidhuber. World models. In *Advances in Neural Information Processing Systems*, pp. 2734–2742, 2018.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. Pmlr, 2018.

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 (ICML)*, pp. 2555–2565, 2019.

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.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Danijar Hafner, Wilson Yan, and Timothy Lillicrap. Training agents inside of scalable world models. *arXiv preprint arXiv:2509.24527*, 2025.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, et al. GAIA-1: A generative world model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Li Hu, Xin Zhang, Li Zhang, et al. Animate anyone: Consistent and controllable image-to-video synthesis. *CVPR*, 2024b.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, et al. DREAMGEN: Unlocking generalization in robot learning through video world models. *arXiv preprint arXiv:2505.12705*, 2025.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR*, 2015.

Po-Chen Ko, Jiayuan Mao, Yilun Du, Shao-Hua Sun, and Joshua B Tenenbaum. Learning to act from actionless videos through dense correspondences. *arXiv preprint arXiv:2310.08576*, 2023.

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview*, 2022. URL `https://openreview.net/forum?id=BZ5a1r-kVsf`.

Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast, and robust structure and motion from casual dynamic videos. In *CVPR*, 2025.

Yunhao Luo and Yilun Du. Grounding video models to actions through goal conditioned exploration. In *International Conference on Learning Representations (ICLR)*, 2025.

Yecheng Jason Ma, Joey Hejna, Chuyuan Fu, Dhruv Shah, Jacky Liang, Zhuo Xu, Sean Kirmani, Peng Xu, Danny Driess, Ted Xiao, et al. Vision language models are in-context value learners. In *The Thirteenth International Conference on Learning Representations*, 2024.

Suraj Nair and Chelsea Finn. Hierarchical foresight: Self-supervised learning of long-horizon tasks via visual subgoal generation. In *International Conference on Learning Representations (ICLR)*, 2020.

James Ni, Zekai Wang, Wei Lin, Amir Bar, Jitendra Malik, et al. From generated human videos to physically plausible robot trajectories. *arXiv preprint arXiv:2512.05094*, 2025.

OpenAI. Video generation models as world simulators. *OpenAI Blog*, 2024.

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.

Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *CVPR*, 2024.

Rolandos Alexandros Potamias et al. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In *CVPR*, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022.

Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. Gaia-2: A controllable multi-view generative world model for autonomous driving. *arXiv preprint arXiv:2503.20523*, 2025.

Oleh Rybkin, Chuning Zhu, Anusha Nagabandi, Kostas Daniilidis, Igor Mordatch, and Sergey Levine. Model-based reinforcement learning via latent-space collocation. In *International Conference on Machine Learning (ICML)*, 2021.

Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. *arXiv preprint arXiv:2502.06764*, 2025.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Basile Terver, Adrien Bardes, Tsung-Yen Yang, Jean Ponce, and Yann LeCun. What drives success in physical planning with joint-embedding predictive world models? *arXiv preprint arXiv:2512.24497*, 2026.

Wan Team. Wan: Open and advanced large-scale video generation models. *arXiv preprint*, 2025.

Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*, 2025.

Ze Yang, Yiming Zhang, Jerry Yu, Junru Cai, and Ming-Hsuan Yang. UniSim: A neural closed-loop sensor simulator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *ICCV*, 2023.

Chuheng Zhang, Tim Pearce, Pushi Zhang, Kaixin Wang, Xiaoyu Chen, Wei Shen, Li Zhao, and Jiang Bian. What do latent action models actually learn? *arXiv preprint arXiv:2506.15691*, 2025.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2025.

Christos Ziakas and Alessandra Russo. Vita: Zero-shot value functions via test-time adaptation of vision-language models. In *International Conference on Learning Representations (ICLR)*, 2026.

A QUALITATIVE EVALUATION AND FAILURE MODES

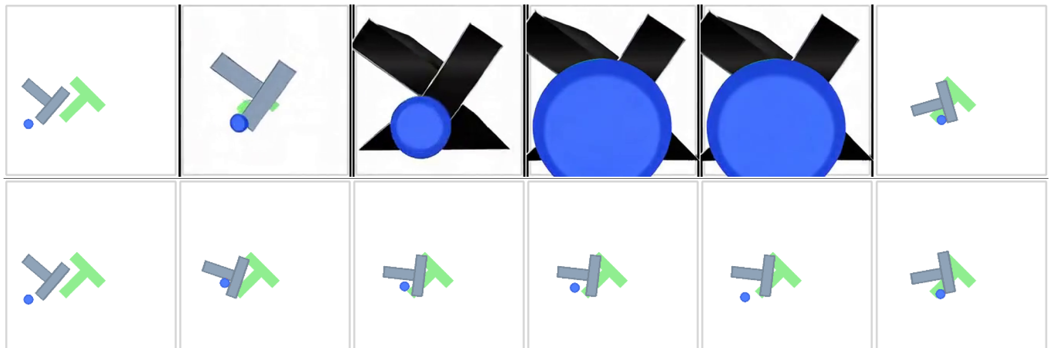


Figure 4: The generated video plan (top) hallucinates large artifacts. GVP-WM (bottom) succeeds.

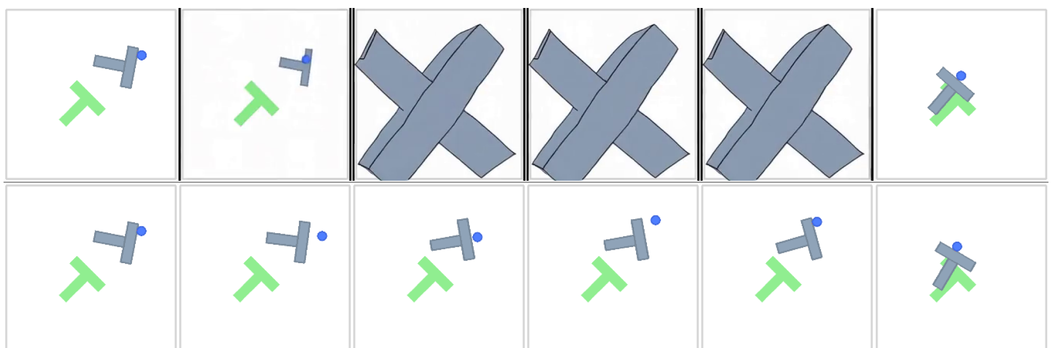


Figure 5: The generated video plan (top) hallucinates large artifacts. GVP-WM (bottom) fails.

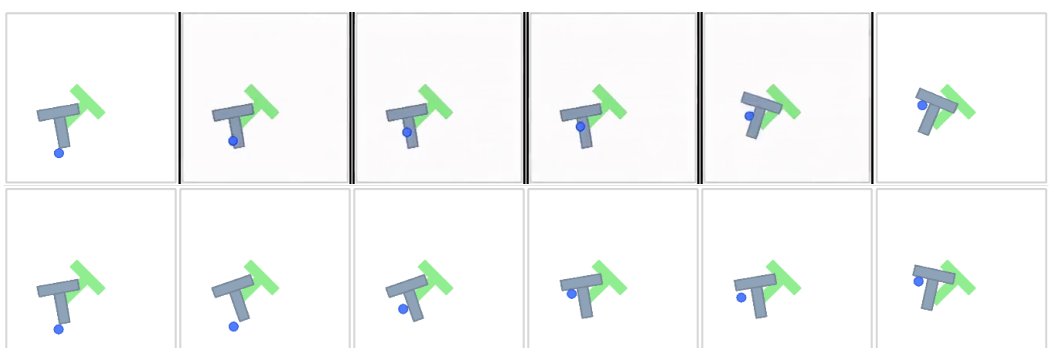


Figure 6: Zero-shot generated video plan (top) violates rigid object physics by passing the agent through T. GVP-WM (bottom) fails.

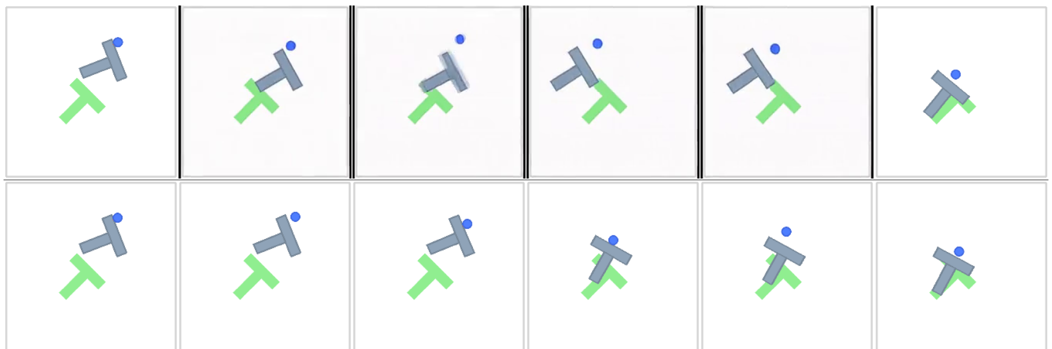


Figure 7: Zero-shot generated video plan (top) violates kinematics by teleporting the agent to goal state. GVP-WM (bottom) fails.

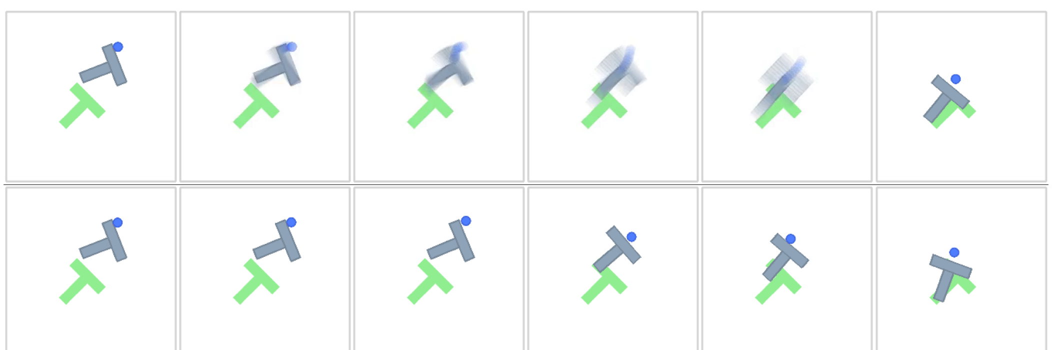


Figure 8: Motion blurred oracle video plan (MB-10) (top). GVP-WM (bottom) succeeds.

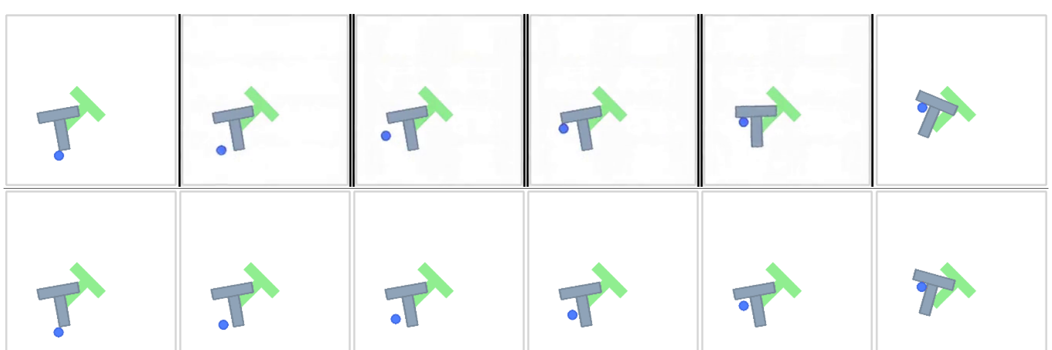


Figure 9: Fine-tuned video model (WAN-FT) generates a physically consistent plan (top). GVP-WM (bottom) succeeds.

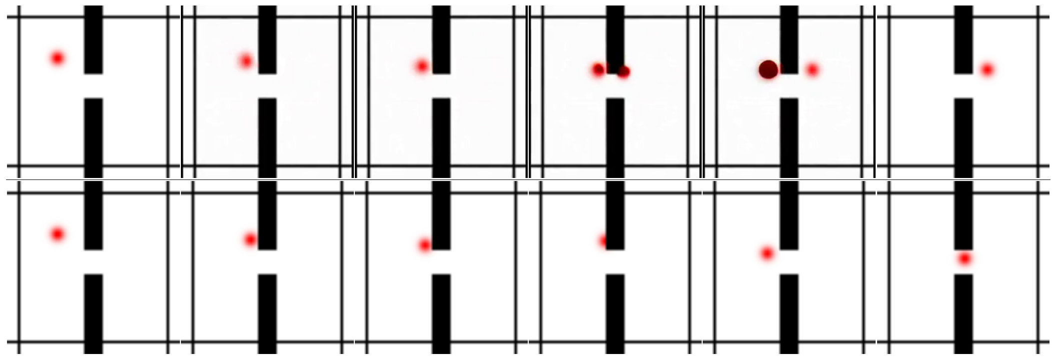


Figure 10: Zero-shot generated video plan (top) exhibits spatial bilocation, depicting the agent at 2 locations. GVP-WM (bottom) fails.

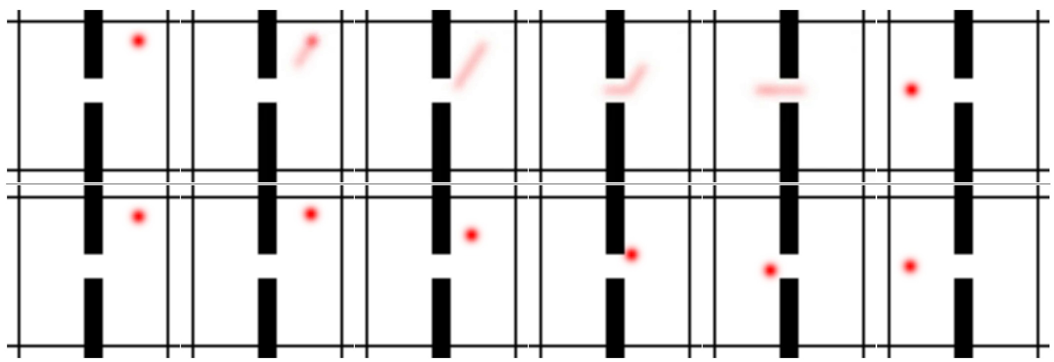


Figure 11: Motion blurred oracle video plan (MB-10) (top). GVP-WM (bottom) succeeds.

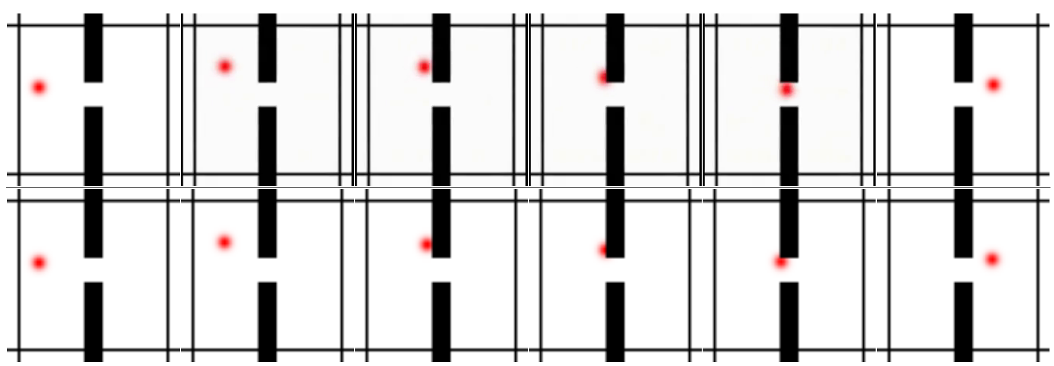


Figure 12: Fine-tuned video model (WAN-FT) generates a physically consistent plan (top). GVP-WM (bottom) succeeds.

Algorithm 2 Grounding Video Plans with World Models

Require: initial state $s_0 = (o_0, p_0)$; goal observation o_g; video model $\mathcal{G}$; world model (E_ϕ, f_ψ); MPC parameters (T, K); ALM iteration and penalty parameters
Generate a video plan: $\tau_{\text{vid}} \sim \mathcal{G}(\cdot \mid o_0, o_g, c)$
Encode video plan into latent space: $z^{\text{vid}}_{0:T} \leftarrow E_\phi(\tau_{\text{vid}})$
Initialize primal variables: $\mathcal{Z} \leftarrow z^{\text{vid}}_{0:T}$, $\mathcal{A} \leftarrow \mathbf{0}$
for $t = 0$ **to** $T - 1$ **step** K **do**
  Set initial latent state of trajectory: $z_t \leftarrow E_\phi(s_t)$
  Initialize dual variables: $\lambda \leftarrow \mathbf{0}$, $\rho \leftarrow \rho_0$
  for $k = 1$ **to** O_{ALM} **do**
    for $i = 1$ **to** I_{ALM} **do**
      Video Guidance: $\mathcal{L}_{\text{vid}}(z_{t+1:T-1}, z^{\text{vid}}_{t+1:T-1})$
      Terminal Latent Goal: $\mathcal{L}_{\text{goal}}(z_T, z_g)$
      Dynamic Constraints:$\mathcal{L}^t_{\text{dyn}}(z_{t-H:t+1}, a_{t-H:t}; f_\psi)$
      Compute Augmented Lagrangian $\mathcal{L}_\rho$ (Eq.4)
      Primal update: $(\mathcal{Z}, \mathcal{A})$ via a gradient step on $\mathcal{L}_\rho$
    end for
    Dual update: $\lambda \leftarrow \lambda + \rho\, \mathcal{L}_{\text{dyn}}$
    Penalty update: $\rho \leftarrow \min(\gamma\rho, \rho_{\max})$
  end for
  Execute $a_{t:t+K}$ from $\mathcal{A}$ with sampling refinement
  Update current state: $s_{t+K} \leftarrow (o_{t+K}, p_{t+K})$
end for

B Algorithm

C Planning Time Comparison

Table 3: Average planning time per episode in seconds.

Method	PushT			Wall	
	T=25	T=50	T=80	T=25	T=50
MPC-CEM	277	596	790	121	198
MPC-GD	96	208	475	44	74
GVP-WM	**86**	**158**	**208**	**24**	**43**

D Ablation Studies

D.1 Effect of Video Guidance

In GVP-WM, video plans are used in two ways: (i) to initialize the latent trajectory and (ii) to provide a scale-invariant video alignment loss weighted by λ_v. Removing video guidance entirely (random latent state initialization with $\lambda_v = 0$) affects performance differently depending on video quality. While removing video guidance improves performance under zero-shot out-of-distribution video plans (WAN-0S), it degrades performance under higher-quality guidance, with success decreasing from 0.82 to 0.68 for domain-adapted generated videos (WAN-FT) and from 0.98 to 0.68 for oracle video plans. Removing latent state initialization from the video plan causes a substantial drop under both domain-adapted and oracle guidance: success decreases from 0.82 to 0.60 with WAN-FT guidance and from 0.98 to 0.62 with oracle video plans, indicating the importance of latent trajectory initialization from video plans. The video alignment loss consistently improves performance across all video sources. Compared to removing the alignment term ($\lambda_v = 0$), adding scale-invariant alignment improves success from 0.54 to 0.56 under zero-shot guidance (WAN-0S), from 0.72 to 0.82 under domain-adapted guidance (WAN-FT), and from 0.92 to 0.98 under oracle video plans.

Table 4: Ablation study on Push-T ($T = 25$). We evaluate the contribution of each component by removing or replacing it. WAN-0S: Zero-shot video, WAN-FT: Fine-tuned video, ORACLE: Ground-truth video.

METHOD	WAN-0S	WAN-FT	ORACLE
Video Guidance			
NO VIDEO GUIDANCE	**0.68**	0.68	0.68
NO VIDEO INIT	0.66	0.60	0.62
NO VIDEO LOSS	0.54	0.72	0.92
Collocation & ALM Solver			
NO COLLOCATION	0.08	0.12	0.60
PROJECTED SGD	0.54	0.72	0.92
Planning & Execution			
OPEN LOOP (NO MPC)	0.44	0.80	0.86
NO LOCAL REFINEMENT	0.52	0.72	0.92
Video Alignment Metric			
MSE ALIGNMENT	0.54	0.64	0.90
GVP-WM	0.56	**0.82**	**0.98**

These gains suggest that alignment is beneficial across settings, with larger improvements when the video guidance is more reliable.

D.2 EFFECT OF LATENT COLLOCATION AND ALM SOLVER

We evaluate the role of latent collocation by fixing latent states to the video plan ($z_t = z_t^{\text{vid}}$) and optimizing only actions. This ablation causes performance to collapse to 0.12 under fine-tuned video guidance, demonstrating that latent trajectories produced by video models are not dynamically feasible and cannot be executed directly. We additionally replace our smooth `tanh` action parameterization with projected gradient descent using action clipping. This substitution reduces success from 0.82 to 0.72, indicating that smooth unconstrained optimization improves convergence of the ALM solver.

D.3 EFFECT OF MPC EXECUTION

To study the execution strategy, we remove receding-horizon control and execute the optimized trajectory in open loop. This reduces performance from 0.82 to 0.80, showing that feedback during execution helps correct model mismatch. Removing the local refinement step further degrades success to 0.72, indicating that this lightweight post-optimization refinement is important for resolving residual inaccuracies left by the non-convex objective.

D.4 EFFECT OF SCALE-INVARIANT ALIGNMENT

Finally, we ablate the video alignment objective by replacing the scale-invariant normalized loss with a standard MSE loss on latent states. This reduces success under fine-tuned video guidance from 0.82 to 0.64, confirming that latent representations from video models exhibit magnitude drift relative to the world model training distribution and that scale invariance is essential for effective grounding.

E IMPLEMENTATION DETAILS

E.1 ENVIRONMENTS AND EVALUATION PROTOCOLS

We evaluate in two control environments that are out-of-distribution relative to the internet-scale pretraining data of the video generator. In both environments, goal states are specified purely as visual observations, and no proprioceptive goal information is available during planning.

Table 5: Dino-WM World Model Architecture and Training Configuration

Category	**Parameter**	**Value**
Architecture	Visual Encoder	DINOv2-ViT-S/14 (frozen)
	Latent Dynamics Predictor	ViT (6 layers, 16 heads, MLP dim 2048)
	Decoder (visualization only)	VQ-VAE (384 channels, 2048 entries)
	Image Resolution	224×224
	History Frames	3
	Prediction Horizon	1 frame
	Frame Skip	5
Training	Training Trajectories	18,500 (Push-T) & 1,920 (Wall)
	Batch Size	32
	Training Epochs	100
	Encoder Learning Rate	1×10^{-6}
	Predictor Learning Rate	5×10^{-4}
	Decoder / Action Encoder LR	3×10^{-4} / 5×10^{-4}

Push-T. Push-T (Chi et al., 2025) is a contact-rich manipulation task in which a circular agent pushes a T-shaped block toward a target configuration in 2D, requiring reasoning about low-level physical interactions and rigid-body contact dynamics. A rollout is considered successful if the final block pose satisfies both positional and rotational accuracy thresholds relative to the target. We evaluate on 50 distinct initial–goal pairs for planning horizons $T \in \{25, 50, 80\}$, where target goals are drawn from held-out expert demonstrations of corresponding lengths. We report Success Rate (SR), defined as the fraction of successful rollouts across all evaluation episodes.

Wall. Wall is a navigation task in which an agent must maneuver around a barrier to reach a target region using abstract 2D visual observations. The task does not involve contact dynamics and primarily tests visual planning and geometric feasibility. Success is defined by the Euclidean distance between the agent's final 2D position and the target location. We evaluate on 50 initial–goal pairs for planning horizons $T \in \{25, 50\}$, using expert demonstrations that traverse the wall in the corresponding number of steps. We report Success Rate (SR), defined as the fraction of successful rollouts across all evaluation episodes.

E.2 World Model

We perform planning using pre-trained, action-conditioned world models from DINO-WM (Zhou et al., 2025). Separate world models are trained for Push-T and Wall, each using data collected within its respective environment. In Push-T, the world model is trained on expert demonstration trajectories, while in Wall the world model is trained from randomly collected trajectories, following the DINO-WM architecture and training. The world model consists of a frozen DINOv2 ViT-S/14 visual encoder, a transformer-based latent dynamics predictor conditioned on actions, and a VQ-VAE decoder used only for visualization. We use the publicly released DINO-WM training setup and checkpoints without modification. The world models are trained offline and are not fine-tuned during evaluation, isolating the contribution of the planning method. The world model architecture and training configuration are summarized in Table 5.

E.3 Video Generation Model

We use the Wan2.1-FLF2V-14B (720p) Wan Team (2025) video generation model, conditioned on the first frame, to synthesize open-loop video plans. To bridge the domain gap between internet-scale video data and robotic environments, we apply supervised fine-tuning (SFT) using Low-Rank Adaptation (LoRA) via the DiffSynth framework. Fine-tuning is performed with rank $r = 32$ for 10 epochs, repeating the dataset 10 times per epoch to reduce data loading overhead. Training uses 4 $\times$ NVIDIA A100 (80GB) GPUs, a learning rate of 1×10^{-5}, and mixed-precision, taking approximately 5 hours and 40 minutes per epoch.

E.4 Spatial and Temporal Alignment

The Push-T environment provides 224×224 square observations, while the Wan2.1 video model expects 1280×720 inputs with a 16:9 aspect ratio. We apply symmetric padding to the initial and goal observations prior to video generation and crop generated frames before world-model encoding. Since the video generator and world model operate at different temporal strides, generated video latents are interpolated or subsampled to match the world-model frame skip.

E.5 Configuration

Planning is performed via video-guided latent collocation with receding-horizon execution ($K = 1$) under an action-conditioned world model. We solve the resulting constrained optimization using a primal–dual augmented Lagrangian method (ALM) with Adam updates, using I_{ALM} inner (primal) iterations per outer (dual) step and O_{ALM} outer iterations in total. Hyperparameters are selected using a small held-out validation set of 20 trajectories and are fixed across all experiments. For Push-T at horizon $T = 25$, we tune the penalty growth factor $\gamma \in \{1.2, 1.3, 1.5, 1.7, 1.9, 2.2, 2.5, 2.9\}$, video alignment weight $\lambda_v \in \{0.01, 0.1, 1.0, 10.0, 20.0, 50.0\}$, action regularization weight $\lambda_r \in \{0.01, 0.05, 0.1, 0.5\}$, goal alignment weight $\lambda_g \in \{0.1, 1.0, 5.0, 10.0, 20.0, 50.0\}$, $I_{\mathrm{ALM}} \in \{10, 25, 50, 100, 200\}$, $O_{\mathrm{ALM}} \in \{5, 10, 25, 50\}$, and learning rate $\eta \in \{0.01, 0.05, 0.1, 0.5, 1.0\}$. The final Push-T configuration uses $\lambda_r = 0.05$, $\lambda_v = 1.0$, $\lambda_g = 10.0$, $I_{\mathrm{ALM}} = 25$, $O_{\mathrm{ALM}} = 25$, $\rho_0 = 1.0$, and $\gamma = 1.9$, with optional local refinement using 500 samples with variance 0.3. For Push-T horizons $T \in \{50, 80\}$, we set $\lambda_r = 0.1$. For Wall, we use the same configuration as Push-T, except setting $\gamma = 1.5$ for $T = 25$; $T = 50$ matches the Push-T configuration.