VIDEAS: DISTILLING EXPLICIT ACTION SEMANTICS FROM DEMONSTRATION VIDEOS FOR WORLD MODELS VIA PRIOR-GUIDED SIMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

World models learn internal representations of environment dynamics to predict future states, enabling agents to optimize action plans without physical interactions. However, developing world models that genuinely internalize underlying causal physical laws to explicitly reason about action preconditions and subsequent state transitions remains an open challenge. In this paper, we propose VIDEAS, a data distillation framework that transforms continuous physical dynamics from operational videos into explicit action semantics for foundation models. Specifically, it deconstructs visual demonstrations into discrete action trajectories and utilizes advanced vision-language models (VLMs) to extract structured knowledge encapsulating action preconditions and effects. To ensure physical consistency, we introduce a prior-guided trajectory simulation mechanism grounded within a text-based environment to rigorously validate the extracted knowledge. Notably, we incorporate negative trajectories to enrich knowledge completeness and enhance data diversity to mitigate cognitive bias. Furthermore, we present VIDEAS-WM, an 8B/9B-parameter suite of language-based world models trained on 34K high-quality samples derived from AgiBot-World dataset. Extensive experiments demonstrate that VIDEAS-WM establishes state-of-the-art performance in high-level embodied action semantic reasoning, exhibiting profound physical understanding and robust generalization across unseen scenarios.

1 INTRODUCTION

World models enable embodied agents to simulate and refine action plans internally, circumventing the costs and risks associated with real-world trial and error (LeCun, 2022; Ha & Schmidhuber, 2018; Chen et al., 2025b). Currently, world models have exhibited substantial potential in facilitating tasks ranging from robotic manipulation and navigation to locomotion and policy learning (Dong et al., 2026). Despite these advances, learning world models with explicit knowledge of action preconditions and effects remains an open challenge (Hu et al., 2025; Xie et al., 2025a), which entails determining the applicability of an action within the current state and predicting the subsequent environment dynamics upon execution.

Existing methods explore world models from various paradigms. Large language models (LLMs) are frequently leveraged as potential world models for planning (Hao et al., 2023) and simulation (Wang et al., 2025b) via prompt strategies; however, their pre-training corpora capture real-world physical laws only implicitly. Consequently, relying solely on prompting to elicit this latent knowledge often results in superficial approximations of environmental transitions and physical inconsistencies during long-horizon rollouts (Wang et al., 2024). Observation-level generative models (Chen et al., 2024; Qin et al., 2025) synthesize high-fidelity future states by directly reconstructing raw visual and geometric observations, but they incur significant computational overhead by predicting numerous task-irrelevant details. Existing latent-space approaches (Assran et al., 2023; 2025) model environmental dynamics within compressed embeddings, which inherently sacrifice interpretability and struggle to preserve fine-grained semantics. Overall, current research often emphasizes prediction accuracy while neglecting the underlying causal dependencies governing action execution.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

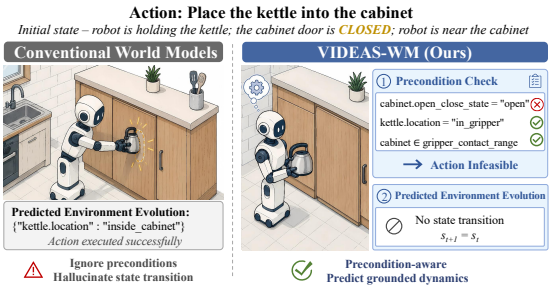


Figure 1: Conventional world models vs. VIDEAS-WM. For an infeasible action, the former ignores action preconditions and hallucinates successful state transitions, while VIDEAS-WM verifies preconditions before predicting environment evolution, correctly judging the action infeasible.

To bridge this gap, we propose **VIDEAS** (**VI**deo-**D**istilled **E**xplicit **A**ction **S**emantics), a novel framework that distills action semantics from in-the-wild operational videos into foundation models, establishing high-level world models equipped with structured knowledge of action preconditions and effects. Inspired by humans developing physical understanding through observing environmental dynamics, we leverage massive online corpora of operational videos that inherently encapsulate the temporal and causal dependencies of actions and the resulting environmental evolution. Rather than superficial video transcription, VIDEAS represents the video demonstrations as discrete action trajectories and leverages advanced VLMs to extract structured knowledge from the multimodal data. To ensure physical correctness, we introduce a trajectory simulation mechanism guided by execution priors to rigorously validate extracted action semantics. This mechanism simulates the given trajectory within a text-based environment to progressively derive the semantics of individual actions, and verifies the simulated outcomes against execution priors, thereby filtering out physically inconsistent knowledge. Furthermore, we deliberately construct failed trajectories to improve the completeness of action precondition knowledge and enhance data diversity. As a further step, we introduce **VIDEAS-WM**, an 8B/9B-parameter suite of language-based world models trained on 34K high-quality samples curated from AgiBot-World dataset (Bu et al., 2025).

We conduct comprehensive experiments to evaluate the capability of VIDEAS-WM in robot behavior simulation and high-level world modeling. Experimental results across multiple benchmarks demonstrate that VIDEAS-WM effectively internalizes embodied action semantics, attaining state-of-the-art performance. Specifically, VIDEAS-WM achieves average performance gains of 10.67% on BTSIMBENCH (Wang et al., 2025b) and 7.00% on LogicEnvEval (Wang et al., 2026) compared to the strong Gemini 3.1 Pro baseline (DeepMind, 2026). Furthermore, VIDEAS-WM exhibits strong cross-scenario generalization on challenging WorldPrediction tasks (Chen et al., 2025a), outperforming larger open-source foundation models.

Overall, our work makes the following key contributions:

- We propose VIDEAS, a scalable data distillation framework that transforms the continuous physical dynamics of operational videos into explicit action semantics for foundation models. It features a prior-guided trajectory simulation mechanism and the incorporation of failed trajectories to ensure semantic correctness and enhance physical grounding.
- Leveraging the VIDEAS framework, we automatically curate a high-quality corpus of 34K samples and present VIDEAS-WM, an 8B/9B-parameter suite of language-based world models.
- Experimental results show that VIDEAS-WM achieves state-of-the-art performance in high-level embodied action semantic reasoning, validating its practical utility on downstream applications including robotic simulation and reasoning.

2 Related Work

World Models. Existing work explores world models through distinct paradigms. Observation-level generative models synthesize future states by reconstructing visual and geometric observations (Yang et al., 2024; Bruce et al., 2024; Bahmani et al., 2024). While high-fidelity, they incur heavy computational overhead and struggle to maintain physical consistency over a long horizon. Latent-space methods (Assran et al., 2023; Garrido et al., 2024) model environmental dynamics within compressed abstract representations, which capture underlying motion and semantic structures but sacrifice interpretability. Reinforcement learning-based approaches, exemplified by the Dreamer series (Hafner et al., 2020; 2025), function as differentiable simulators to learn policies through imagined rollouts. Recently, Cosmos 3 (Aditi et al., 2026) presents an open-weight family of omnimodal

world models that jointly model language, image, video, audio, and action, supporting multimodal reasoning, world simulation, and action generation within a unified architecture. Additionally, some work (Feng et al., 2025; Wang et al., 2025d) factorizes the environment into interacting entities to capture object-level dynamics. Another active line (Hao et al., 2023; Wang et al., 2025c; Li et al., 2025) uses LLMs to model abstract transitions through natural language, yielding greater interpretability and computational efficiency. However, relying on implicit physical knowledge from pre-training renders their long-horizon predictions brittle (Wang et al., 2024). Moreover, existing methods primarily focus on future state prediction but overlook assessing action feasibility before execution—a gap that our work is designed to address.

Learning from Demonstration Videos. Demonstration videos naturally encapsulate physical interactions and causal dynamics, serving as a scalable resource for robotics learning. One prominent direction (Chen et al., 2025c; Li et al., 2024; Bharadhwaj et al., 2024) focuses on learning low-level visuomotor policies from video data. For instance, LAPA (Ye et al., 2025b) infers discrete latent actions via unsupervised pre-training, bridging actionless internet video and policy learning. Parallel efforts (Wang et al., 2025a; Xie et al., 2025b) transform operational videos into code-based policies for high-level task planning. To capture explicit causal logic, some work (Huang et al., 2025; Ye et al., 2025a) derives formal symbolic domains from demonstrations, which inherently restricts their open-world scalability. VLWM (Chen et al., 2025b) learns world models on natural videos by using natural language as a descriptive abstraction to represent action-conditioned state transitions. In contrast, we distill real-world demonstration videos into structured action semantics, explicitly formulating action preconditions and effects to equip world models with grounded physical logic.

3 METHOD

3.1 PROBLEM FORMULATION

In this work, we learn a high-level embodied world model f_θ to internalize explicit action semantics. Given the current environment state s_t and an intended robot action a_t, the model deduces a set of requisite preconditions p_t for the action, and predicts the state transitions Δs_t upon execution:

$$\begin{aligned} p_t &\sim f_\theta\big(\cdot \mid \text{config}_{\text{pre}}, s_t, a_t\big), \\ \Delta s_t &\sim f_\theta\big(\cdot \mid \text{config}_{\text{eff}}, s_t, a_t\big), \end{aligned} \tag{1}$$

where config represents the system prompt, s_t denotes the text-based environment observation, a_t denotes the textual action instruction, and p_t denotes the set of textual action preconditions. The subsequent state is then given by $s_{t+1} = s_t \oplus \Delta s_t$, where $\oplus$ applies the transitions to s_t.

Furthermore, to determine the action executability, the model must ground these preconditions in the concrete environmental context. This process entails assessing whether an individual condition $c \in p_t$ is satisfied under the current state s_t:

$$v \sim f_\theta\Big(\cdot \mid \text{config}_{\text{grd}}, s_t, c\Big), \tag{2}$$

where $v \in \{\text{True}, \text{False}\}$ is a boolean indicator reflecting the fulfillment of the condition.

3.2 DATA PREPROCESSING

Learning a physically-grounded embodied world model requires a high-quality corpus of action semantics that encompasses causal knowledge of action preconditions and effects. As illustrated in Figure 2, we present our data preprocessing pipeline based on the AgiBot-World dataset (Bu et al., 2025), which comprises 168K long-horizon manipulation episodes accompanied by language annotations for both the overall tasks and individual action steps.

Given the abundance of semantically duplicated episodes in the raw dataset, we initially filter it to isolate a core set of unique manipulation trajectories. Leveraging the original frame-level demarcations provided for each action step, we segment the continuous demonstrations into video clips with consistent motion and single semantic behavior. However, these clips are highly heterogeneous in visual presentation and annotation quality. Specifically, the raw language annotations often exhibit incomplete descriptions and contextual inconsistencies across action steps. Thus, we employ FlowNet (Dosovitskiy et al., 2015) to prune clips with indistinct motion, and utilize VLMs to

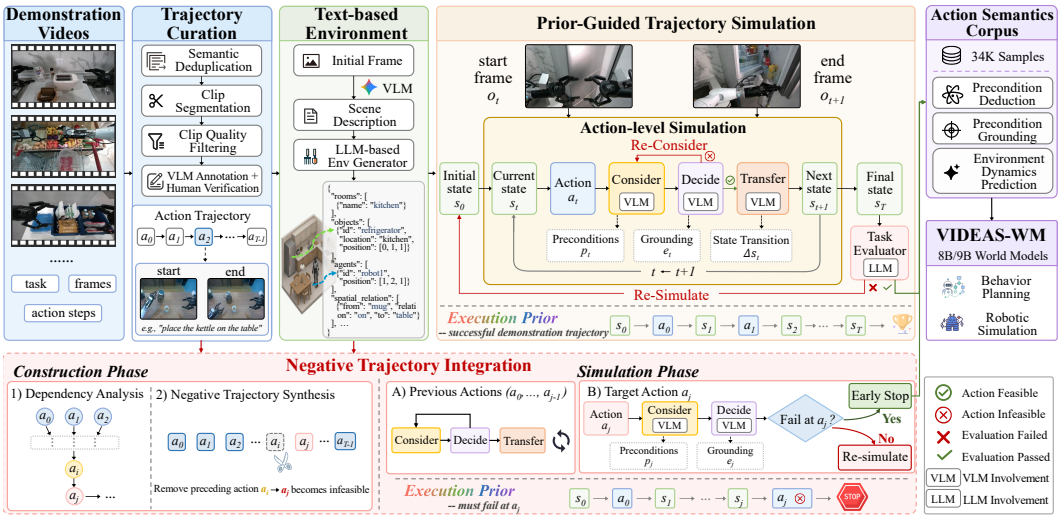


Figure 2: Overview of VIDEAS. Based on in-the-wild demonstration videos, VIDEAS first curates them into discrete action trajectories and grounds them into text-based structured environments, where it leverages VLMs to iteratively simulate each action via *Consider–Decide–Transfer* phases under execution priors. Moreover, it constructs negative trajectories via dependency-aware mutation to enrich precondition knowledge. Finally, the distilled 34K-sample corpus serves to train VIDEAS-WM, an 8B/9B-parameter suite of language-based world models.

reformulate precise action descriptions for valid clips. We further incorporate human-in-the-loop verification to enforce semantic fidelity, guaranteeing alignment between the textual representations and visual dynamics. Through this process, the raw demonstrations are transformed into discrete, logically coherent action trajectories $\tau = (a_0, a_1, \ldots, a_{T-1})$, where T is the trajectory length.

3.3 Action Semantics Distillation

To distill continuous physical dynamics into structured action semantics, we introduce a prior-guided trajectory simulation mechanism established within text-based environments. The mechanism progressively dissects individual embodied actions and grounds the visual demonstrations in a deterministic textual context, yielding a fully observable state representation. This enables the precise tracking of intermediate state transitions and supports the explicit translation of action preconditions into checkable textual constraints. Furthermore, language inherently abstracts away the low-level details of complex physical interactions, allowing the model to focus on the high-level causal logic that governs action preconditions and environmental transitions.

Text-Based Environment Generation. For each action trajectory, VIDEAS constructs a structured, text-based simulated environment. Specifically, it employs VLMs to extract a comprehensive scene description from the initial video frame, and then utilizes LogicEnvGen (Wang et al., 2026) to generate a JSON-formatted environment from the description. This environment specifies the foundational configurations, such as room layouts, doors, and windows. Within the environment, the physical entities, including robotic agents and manipulated objects, are characterized by fine-grained properties, encompassing text descriptions, 3D sizes, positions, and entity-specific attributes (e.g., *open_close_state, indication_reading*). To fully contextualize the physical world, the generated environment further captures the relations among entities, including spatial relations (e.g., *on, in*) and non-spatial relations (e.g., *hold, grasp*).

Prior-Guided Trajectory Simulation. Building upon the initial simulated environment, VIDEAS leverages advanced VLMs to simulate the action trajectory τ within a textual space, thereby incrementally distilling explicit action semantics. Following BeSimulator (Wang et al., 2025b), the execution of each atomic action a_t is modeled from the current state s_t through a fine-grained three-phase process: *consider*, *decide*, and *transfer*. To prevent textual hallucinations and ensure physical consistency, the simulation is grounded by the visual observations before and after the action execution (i.e., the start frame o_t and the end frame o_{t+1}), providing reliable visual evidence of the

execution context. During the first two phases, VIDEAS evaluates whether the action execution aligns with real-world physical logic . Specifically, in the **consider** phase, it extracts the necessary preconditions for successful execution:

$$p_t = \mathcal{M}^{\text{con}}(s_t, a_t, o_t, o_{t+1}), \tag{3}$$

where $\mathcal{M}$ denotes the teacher VLM invoked via greedy decoding (temperature = 0) during distillation, distinct from the target world model f_θ. Subsequently, in the **decide** phase, VIDEAS assesses the satisfaction of each precondition $c \in p_t$ against the textual world state and aggregates the results into an executability judgment:

$$e_t = \bigwedge_{c \in p_t} v_c, \quad v_c = \mathcal{M}^{\text{dec}}(s_t, c), \tag{4}$$

where $v_c \in \{\text{True}, \text{False}\}$ denotes the predicted satisfaction of condition c, and $e_t = \text{True}$ indicates that a_t is feasible under s_t. Upon validating the feasibility, VIDEAS models the environment evolution in the **transfer** phase:

$$\Delta s_t = \mathcal{M}^{\text{tra}}(s_t, a_t, o_t, o_{t+1}), \tag{5}$$

where Δs_t captures the state transitions induced by the action.

However, directly relying on the unconstrained inferences of VLMs sometimes yields physically inconsistent or incomplete action semantics. To address this, we utilize the execution priors of the action trajectories, which originate from successful real-world demonstrations. Since these trajectories naturally guarantee the successful execution of all actions and the task completion, we use this prior to rigorously validate the extracted action semantics. Specifically, VIDEAS uses a dual-grained iterative verification mechanism at both the action and trajectory levels. At the action level, if an action is deemed infeasible (i.e., $e_t = \text{False}$), VIDEAS first postulates that this anomaly stems from misidentified preconditions, and then activates a **re-consider** phase to guide VLMs in rectifying the precondition knowledge. If the action remains unexecutable after VLM reflection, which indicates incorrect state transitions in the preceding actions, VIDEAS restarts the simulation. At the trajectory level, after the entire trajectory simulation, VIDEAS leverages LLMs to evaluate the task completion based on the final state s_T. If completed, which indicates that the simulation is consistent with the prior, VIDEAS collects the action semantics distilled from the trajectory. Conversely, completion failure reveals flawed state transitions during simulation, leading VIDEAS to conduct diagnostics and integrate the feedback into the next simulation iteration. This process repeats until a successful simulation or a predefined iteration limit.

Negative Trajectory Integration. Learning exclusively from successful demonstrations inherently restricts the world model to observing satisfied preconditions. This hinders the model from generalizing the learned causal knowledge to evaluate action feasibility when physical preconditions are unmet. Therefore, we introduce negative action trajectories, which fail at the predetermined step due to unfulfilled preconditions. This integration enriches the semantic completeness of precondition knowledge and provides diverse negative samples for action precondition assessment.

To construct negative trajectories, we mutate the successful action trajectories based on the action execution dependencies. Specifically, we first define a set of common dependency types $\mathcal{D}$, including possession dependencies (e.g., *requiring a previously grasped object*) and state dependencies (e.g., *relying on a previously opened cabinet*). We then employ advanced LLMs to analyze the causal relationships among actions. This yields a dependency record associating each action a_k with the preceding actions it depends on:

$$\mathcal{P}_k = \{\, a_i \in \tau \mid i < k \text{ and } \exists\, d \in \mathcal{D} \text{ s.t. } a_i \xrightarrow{d} a_k \,\}, \tag{6}$$

where $a_i \xrightarrow{d} a_k$ denotes that the execution of a_k depends on a_i through a dependency of type d. We then automatically generate a negative trajectory τ^- by removing a preceding action a_i from the trajectory τ, rendering the execution of the target action a_j with $j = \min\{\, k \mid a_i \in \mathcal{P}_k \,\}$ infeasible.

For a negative trajectory τ^-, VIDEAS simulates it under the execution prior that it must fail exactly at the predetermined action a_j. In detail, VIDEAS simulates the preceding action sequence normally until reaching a_j, where it performs only the consider and decide phases, relying purely on the textual state with visually mismatched frames masked. If a_j is correctly judged as unexecutable,

which indicates that the VLM extracts the requisite preconditions and accurately identifies their non-fulfillment within the current environment, VIDEAS triggers an early stop and collects the derived knowledge as negative samples. In contrast, an erroneous evaluation of a_j as executable violates the negative prior and forces VIDEAS to restart the simulation.

3.4 Model Training

We frame the training stage as optimizing the foundation model f_θ to internalize the structured action semantics. Formally, we construct a consolidated training dataset $\mathcal{C} = \mathcal{C}_{\text{pre}} \cup \mathcal{C}_{\text{grd}} \cup \mathcal{C}_{\text{eff}}$ spanning three learning tasks, namely precondition deduction, precondition grounding, and environment dynamics prediction, whose input-output pairs are instantiated as

$$\begin{aligned} \mathcal{C}_{\text{pre}} &: \big((\text{config}_{\text{pre}}, s_t, a_t),\ p_t\big), \\ \mathcal{C}_{\text{grd}} &: \big((\text{config}_{\text{grd}}, s_t, c),\ v_c\big), \\ \mathcal{C}_{\text{eff}} &: \big((\text{config}_{\text{eff}}, s_t, a_t),\ \Delta s_t\big). \end{aligned} \tag{7}$$

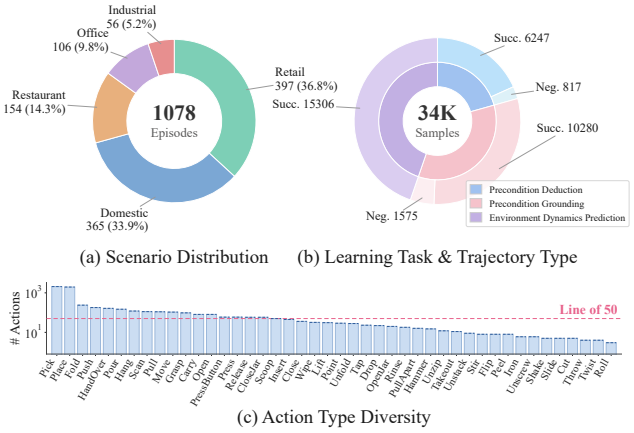
We unify these tasks into a standardized sequence generation paradigm and optimize θ with the autoregressive objective:

$$\mathcal{L}(\theta) := -\mathbb{E}_{(x,y)\sim\mathcal{C}} \sum_{l=1}^{|y|} \log P(y_l \mid x, y_{<l}; \theta). \tag{8}$$

4 Experiments

4.1 Experimental Setup

Dataset Statistics. In our experiments, we utilize the AgiBot-World dataset (Bu et al., 2025) as the demonstration data source, which contains extensive long-horizon manipulation episodes spanning diverse daily scenarios (e.g., *domestic, retail, and industrial environments*). Through rigorous deduplication and quality filtering, we extract a core subset of 1078 unique episodes covering 215 tasks and 86 action types, as illustrated in Figure 3. Ultimately, we construct a high-quality corpus comprising 34K data samples, which capture explicit action semantics.



(a) Scenario Distribution (b) Learning Task & Trajectory Type

(c) Action Type Diversity

Figure 3: Training corpus statistics. In (b), Succ. and Neg. denote samples derived from successful and negative trajectories respectively. (c) shows representative action types.

Implementation Details. We employ Gemini 3.1 Pro (DeepMind, 2026) as the teacher model $\mathcal{M}$, and adopt Llama-3.1-8B (Grattafiori et al., 2024), Qwen3-8B (Qwen Team, 2025), and Qwen3.5-9B (Qwen Team, 2026) as the base models to instantiate VIDEAS-WM. We perform LoRA fine-tuning for 3 epochs and the training process is conducted on 4 NVIDIA A800 80GB PCIe GPUs. Detailed training settings are provided in the Appendix B.

Baselines and Benchmarks. To comprehensively evaluate the performance of VIDEAS-WM, we compare it against leading closed-source and open-source models across three benchmarks. Detailed simulation protocols are provided in the Appendix C.

- **BTSIMBENCH** (Wang et al., 2025b): It measures how well a model can capture execution preconditions and predict action outcomes through robotic policy simulation. It comprises 75 robotic policies (i.e., behavior trees) categorized into Good, Counterfactuals, and Unreachable. We adopt

Table 1: Performance (%) on BTSIMBENCH and LogicEnvEval benchmarks. Each policy category contains 25 samples. Gray and green rows represent closed-source baselines and VIDEAS-WM, respectively. The best performance among open-source models is in **bold**.

Model	BTSIMBENCH					LogicEnvEval					
	Delivery	Good	CFactuals	Unreach	Avg	Delivery	Good	CFactuals	Unreach	LBranch	Avg
Closed-Source Models											
Gemini-3.1-Pro	100.00	92.00	64.00	96.00	84.00	100.00	92.00	72.00	92.00	96.00	88.00
Gemini-3-Flash	100.00	88.00	64.00	88.00	80.00	100.00	76.00	76.00	88.00	96.00	84.00
Open-Source Models											
DeepSeek-V4-Pro	100.00	44.00	88.00	76.00	69.33	100.00	52.00	84.00	88.00	96.00	80.00
DeepSeek-V4-Flash	98.67	44.00	80.00	72.00	65.33	100.00	60.00	76.00	84.00	**100.00**	80.00
Gemma4-31B	100.00	76.00	72.00	92.00	80.00	100.00	88.00	84.00	**92.00**	88.00	88.00
Llama3.3-70B	98.67	20.00	92.00	28.00	46.67	99.00	32.00	76.00	48.00	92.00	62.00
Qwen-3.5-27B	98.67	40.00	88.00	76.00	68.00	100.00	72.00	80.00	88.00	**100.00**	85.00
Qwen3-32B	98.67	48.00	72.00	80.00	66.67	76.00	12.00	64.00	48.00	84.00	52.00
Qwen3-14B	98.67	28.00	84.00	52.00	54.67	97.00	32.00	72.00	60.00	96.00	65.00
Qwen2.5-72B	100.00	28.00	84.00	68.00	60.00	96.00	40.00	68.00	64.00	92.00	66.00
Llama3.1-8B	78.67	0.00	76.00	8.00	28.00	88.00	4.00	88.00	16.00	84.00	48.00
VIDEAS-WM-L8B	100.00	**92.00**	**96.00**	**96.00**	**94.67**	100.00	92.00	96.00	88.00	84.00	90.00
Qwen3-8B	85.33	12.00	80.00	32.00	41.33	95.00	16.00	72.00	40.00	96.00	56.00
VIDEAS-WM-Q8B	100.00	76.00	**96.00**	88.00	86.67	97.00	**96.00**	96.00	**92.00**	84.00	92.00
Qwen3.5-9B	86.67	32.00	72.00	52.00	52.00	87.00	28.00	64.00	64.00	96.00	63.00
VIDEAS-WM-9B	100.00	**92.00**	**96.00**	**96.00**	**94.67**	100.00	88.00	**100.00**	**92.00**	**100.00**	**95.00**

the delivery rate and the simulation accuracy across these policy types as the evaluation metrics. The delivery rate evaluates whether the model successfully delivers the final simulation result within a predefined iteration limit.

- **LogicEnvEval** (Wang et al., 2026): It challenges the model to determine action executability and reason about environment dynamics across long-horizon household tasks. Similarly, we conduct evaluations through robot behavior simulation on 100 agent policies from this benchmark. These policies span four categories, including the aforementioned three types and the LackBranch type.
- **WorldPrediction-WM** (Chen et al., 2025a): It assesses whether the model comprehends the causalities of semantically and temporally high-level actions in daily human activities through question answering. Due to original data availability, we utilize a subset of 637 accessible task samples sourced from the COIN (Tang et al., 2019), CrossTask (Zhukov et al., 2019), EPIC-KITCHENS-100 (Damen et al., 2022), and IKEA-ASM datasets (Ben-Shabat et al., 2021).

4.2 Model Performance

Table 1 presents the performance comparison between VIDEAS-WM and baselines on BTSIMBENCH and LogicEnvEval benchmarks. The results show that VIDEAS-WM establishes state-of-the-art performance among open-source solutions, effectively assimilating explicit action semantics to achieve a deeper understanding of high-level physical logic.

Notably, our VIDEAS-WM-9B achieves average scores of 94.67% on BTSIMBENCH and 95.00% on LogicEnvEval, outperforming the strongest open-source model Gemma4-31B by 14.67% and 7.00%, respectively. Moreover, the three VIDEAS-WM models maintain a substantial lead over their corresponding base models across most evaluation metrics, especially regarding the simulation accuracy on Good and Unreachable policies. For instance, on the LogicEnvEval benchmark, our VIDEAS-WM-L8B achieves 92.00% accuracy in simulating and evaluating Good policies, while the Llama3.1-8B yields 4.00% accuracy. This highlights the superiority of VIDEAS-WM in effectively modeling environment dynamics. Furthermore, the performance improvement on the Counterfactuals category demonstrates the efficacy of our models in reasoning about execution preconditions and determining action executability, thereby facilitating the identification of faulty agent policies. These results validate the practical utility of VIDEAS-WM for high-level robotic behavior simulation.

4.3 Cross-Scenario Generalization

To evaluate the cross-scenario generalization capability of VIDEAS-WM, we conduct experiments on the WorldPrediction-WM benchmark. It encompasses diverse everyday activities (e.g., *household chores, furniture assembly, and nursing & caring*). This benchmark requires models to deduce the

Table 2: Performance (%) on WorldPrediction-WM benchmark. Green rows represent our VIDEAS-WM. The best performance within each category (VLMs and Socratic LLMs) is in **bold**.

	Model	WorldPrediction-WM				
		COIN	CrossTask	IKEA-ASM	EPIC-KITCHENS-100	Overall
VLMs	InternVL3.5-38B	47.69	55.56	40.88	**73.44**	54.87
	InternVL3.5-14B	47.18	45.56	35.85	51.56	45.44
	InternVL3.5-8B	45.13	44.44	32.08	40.63	40.41
	Qwen3.5-27B	51.28	55.56	**47.80**	67.19	55.82
	Qwen3-VL-32B	**53.85**	54.44	45.28	69.27	56.45
	Qwen3-VL-8B	51.79	55.56	43.40	67.71	55.03
	Qwen3.5-9B	45.13	56.67	34.59	66.67	50.63
	VIDEAS-WM-9B	53.33	**60.00**	45.28	**73.44**	**58.33**
Socratic LLMs	DeepSeek-V4-Pro	63.08	80.00	32.70	**79.27**	62.79
	DeepSeek-V4-Flash	62.56	**81.11**	28.30	**79.27**	61.70
	MiniMax-M2.7	55.90	71.11	28.93	68.39	55.10
	Llama3.3-70B	51.79	76.67	10.69	62.69	48.35
	Qwen3-32B	57.95	80.00	35.22	75.65	60.75
	Qwen3-14B	58.97	74.44	31.45	74.61	59.03
	Qwen2.5-72B	57.95	**81.11**	28.30	75.13	59.03
	Llama3.1-8B	54.36	74.19	24.53	68.39	54.06
	VIDEAS-WM-L8B	56.92	75.56	38.36	75.13	60.43
	Qwen3-8B	57.44	70.97	32.08	75.65	58.59
	VIDEAS-WM-Q8B	**65.64**	76.67	**41.51**	73.06	**63.42**

correct action responsible for the environment transition between the provided initial and final states. Table 2 shows that VIDEAS-WM-9B attains 58.33% overall accuracy in the VLMs category, outperforming larger open-source models, such as Qwen3.5-27B and InternVL3.5-38B. In the Socratic LLMs category, VIDEAS-WM-Q8B improves over its base model Qwen3-8B by 4.83% and further exceeds DeepSeek-V4-Pro. These results indicate that VIDEAS-WM abstracts universal physical logic and robustly transfers this knowledge to novel domains beyond the training distribution.

4.4 Ablation Studies and Analysis

We conduct ablation studies on the BTSIMBENCH benchmark to evaluate the impact of the two core components of VIDEAS, and the scaling of training data.

Impact of Prior-Guided Trajectory Simulation Mechanism. To assess the efficacy of the prior-guided trajectory simulation mechanism, we construct a naive distillation baseline. This variant directly extracts action semantics from the raw teacher annotations without the iterative verification process. We then fine-tune the three evaluated base models on this naive dataset to observe the performance variations. As shown in Table 3, the ablation variant improves the average performance over the base models by 16.00% to 36.00% across all three architectures. However, our analysis reveals that the semantic knowledge distilled directly from the teacher model suffers from physical inconsistencies and biases (e.g., *missing or incorrect action preconditions, and incomplete state transitions*). Consequently, this ablation variant consistently underperforms VIDEAS-WM, exhibiting an average accuracy gap ranging from 22.67% to 30.67%. This highlights the vital role of our mechanism in ensuring knowledge correctness and providing high-quality supervisory signals.

Table 3: Results of ablation study on the prior-guided trajectory simulation mechanism (Naive Distillation) and the negative trajectory integration (W/O NTI).

Model	Method	BTSIMBENCH			
		Good	CFactuals	Unreach	Avg
Llama3.1-8B	Base	0.00	76.00	8.00	28.00
	Naive Distillation	72.00	32.00	88.00	64.00
	W/O NTI	88.00	28.00	**100.00**	72.00
	VIDEAS-WM	**92.00**	**96.00**	96.00	**94.67**
Qwen3-8B	Base	12.00	80.00	32.00	41.33
	Naive Distillation	52.00	44.00	76.00	57.33
	W/O NTI	68.00	44.00	84.00	65.33
	VIDEAS-WM	**76.00**	**96.00**	**88.00**	**86.67**
Qwen3.5-9B	Base	32.00	72.00	52.00	52.00
	Naive Distillation	80.00	52.00	84.00	72.00
	W/O NTI	**92.00**	48.00	92.00	77.33
	VIDEAS-WM	**92.00**	**96.00**	**96.00**	**94.67**

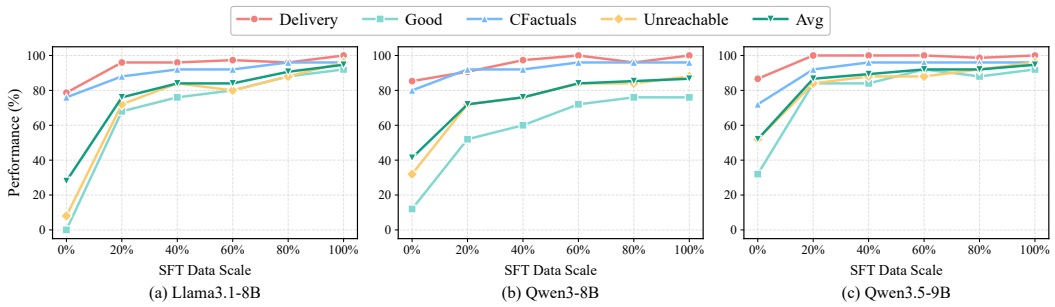


Figure 4: Ablation study results on training data scales for the BTSIMBENCH benchmark.

Impact of Negative Trajectory Integration. To evaluate the effectiveness of incorporating negative trajectories, we construct a data subset by excluding 2.4K negative samples, which exclusively contains the distilled knowledge from successful trajectories. Table 3 compares models trained on this subset and the full dataset. The results reveal a decline in accuracy across most categories for all three models, particularly in the Counterfactuals category. For Qwen3.5-9B, the accuracy of this category drops from 96.00% to 48.00%, which even underperforms the base model. Essentially, the Counterfactuals category evaluates the capability of the model to identify unmet execution preconditions. Thus, exposure to merely positive instances induces severe cognitive bias, causing the model to overfit and blindly predict successful state transitions. This confirms the significance of negative trajectories in enhancing data diversity and knowledge completeness, which enables the world model to develop more comprehensive physical understanding of action execution.

Data Scaling for Model Fine-Tuning. To investigate the influence of training data scale, we uniformly sample 20%, 40%, 60%, and 80% of the full training corpus to perform supervised fine-tuning (SFT) on the three base models. To ensure consistency, we employ a fixed random seed to shuffle the dataset and extract prefixes of the same permutation, guaranteeing that smaller subsets are strict subsets of the larger ones. As shown in Figure 4, the performance scales consistently with the dataset size before reaching a plateau. For instance, fine-tuning the Llama3.1-8B model with merely 20% of the data yields a substantial performance leap, improving the average accuracy from 28.00% to 76.00%. Further increasing the data scale to 100% pushes the average accuracy to 94.67%. This saturation trend suggests that while our dataset effectively injects action semantics into the models, further performance gains through simply scaling the supervised data become marginal.

Human Verification for Distilled Corpus. We conduct human verification of the knowledge extracted by VIDEAS. Specifically, we randomly sample 135 action instances from the 34K corpus and have three independent annotators assess whether the distilled action preconditions and effects align with real-world physical logic. Final labels are determined via majority voting, and we observe a high inter-annotator agreement. The evaluation reveals accuracies of 97.78% for preconditions and 98.52% for effects. We analyze these samples and find that the few invalid precondition samples primarily suffer from missing necessary conditions. For example, the action of placing a scouring pad onto a counter verifies whether the robot holds the pad but misses checking whether the counter is within the gripper contact range. This observation further corroborates the necessity of negative trajectory integration in improving the completeness of precondition knowledge. Meanwhile, the few invalid effect samples mainly result from incomplete state transition predictions.

5 Conclusion

We propose VIDEAS, a VLM-powered data distillation framework that transforms continuous physical dynamics from demonstration videos into explicit action semantics for world models. To ensure the logical correctness and completeness of the extracted semantics, we introduce a prior-guided trajectory simulation mechanism and construct negative trajectories. Furthermore, we present VIDEAS-WM, a suite of language-based world models. Experimental results across multiple benchmarks show that VIDEAS-WM achieves superior performance in embodied action semantic reasoning, exhibiting deep physical understanding and robust generalization in novel scenarios. Our work provides new insights for internalizing explicit physical causalities within world models to facilitate high-level planning and robotic simulation.

AI USE STATEMENT

In this work, we used generative AI tools for generating synthetic data sets. We have not used generative AI tools for designing research methodology, implementing methods, assisting with translation, cleaning datasets, supporting qualitative data analysis, or interpreting results. The remaining disclosure tasks including helping develop theoretical models, formulating mathematical claims, providing critical ingredients for mathematical proofs, assisting in proof writing, and proposing hypotheses are not applicable to this work. Additionally, we used generative AI tools for creating scientific figures and editing the research paper to improve readability. We have reviewed all AI-assisted work. The employment of VLMs constitutes a step in our data distillation framework to prepare data for post-training. Furthermore, we conduct human verification on the knowledge distilled by the VLMs and report the detailed quantitative results in the main text. All authors verify the edited text and modified figures to preserve the original scientific meaning and guarantee overall accuracy. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The development of VIDEAS-WM strictly adheres to established ethical guidelines regarding data usage and model deployment. All raw operational videos and textual annotations originate from publicly available datasets under permissive licenses. For human-in-the-loop verification, we ensure that all annotators receive fair compensation and participate voluntarily with informed consent. Our research exclusively utilizes open-source foundation models to promote methodological transparency and ensure scientific reproducibility. While VIDEAS-WM advances embodied action semantic reasoning, we explicitly restrict its intended use to academic research. Users must be aware that the model may exhibit hallucinations and occasionally generate physically misleading outcomes. Therefore, we urge that applications of this model undergo safety checks to prevent unpredictable hazards.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our work, we describe the data preprocessing pipeline and the trajectory simulation mechanism in Section 3. The exact prompt templates used for this mechanism are provided in Appendix F. We specify the model training setups and essential hyperparameters in Section 4.1 and Appendix B. The detailed evaluation protocols for all benchmarks are available in Appendix C. Moreover, We submit the source code of the VIDEAS framework as supplementary material to facilitate future research.

REFERENCES

Aditi, Niket Agarwal, Arslan Ali, Jon Allen, Martin Antolini, Adeline Aubame, Alisson G. Azzolini, Junjie Bai, Maciej Bala, Yogesh Balaji, Josh Bapst, Aarti Basant, Mukesh Beladiya, Mohammad Qazim Bhat, Zaid Pervaiz Bhat, Dan Blick, Vanni Brighella, Han Cai, Tiffany Cai, Eric Cameracci, Jiaxin Cao, Yulong Cao, Mark Carlson, Carlos Casanova, Ting-Yun Chang, Yan Chang, Yu-Wei Chao, Prithvijit Chattopadhyay, Roshan Chaudhari, Chieh-Yun Chen, Junyu Chen, Ke Chen, Qizhi Chen, Wenkai Chen, Xiaotong Chen, Yu Chen, An-Chieh Cheng, Click Cheng, Xiu Chia, Jeana Choi, Chaeyeon Chung, Wenyan Cong, Yin Cui, Magdalena Dadela, Nalin Dadhich, Wenliang Dai, Joyjit Daw, Alperen Degirmenci, Rodrigo Vieira Del Monte, Robert Denomme, Sameer Dharur, Marco Di Lucca, Ke Ding, Wenhao Ding, Yifan Ding, Yuzhu Dong, Nicole Drumheller, Yilun Du, Aigul Dzhumamuratova, Aleksandr Efitorov, Hamid Eghbalzadeh, Naomi Eigbe, Imad El Hanafi, Hassan Eslami, Benedikt Falk, Jiaojiao Fan, Jim Fan, Amol Fasale, Sergiy Fefilatyev, Liang Feng, Francesco Ferroni, Sanja Fidler, Xiao Fu, Vikram Fugro, Prashant Gaikwad, TJ Galda, Katelyn Gao, Yihuai Gao, Wenhang Ge, Sreyan Ghosh, Arushi Goel, Vivek Goel, Akash Gokul, Rama Govindaraju, Jinwei Gu, Miguel Guerrero, Elfie Guo, Aryaman Gupta, Siddharth Gururani, Hugo Hadfield, Song Han, Ankur Handa, Zekun Hao, Mohammad Harrim, Ali Hassani, Nathan Hayes-Roth, Yufan He, Chris Helvig, and Cyrus Hogg.

Cosmos 3: Omnimodal world models for physical AI. *CoRR*, abs/2606.02800, 2026. doi: 10.48550/ARXIV.2606.02800. URL https://doi.org/10.48550/arXiv.2606.02800.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael G. Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 15619–15629. IEEE, 2023. doi: 10.1109/CVPR52729.2023.01499. URL https://doi.org/10.1109/CVPR52729.2023.01499.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew J. Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *CoRR*, abs/2506.09985, 2025. doi: 10.48550/ARXIV.2506.09985. URL https://doi.org/10.48550/arXiv.2506.09985.

Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas J. Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B. Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 7996–8006. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00764. URL https://doi.org/10.1109/CVPR52733.2024.00764.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *CoRR*, abs/2502.13923, 2025. doi: 10.48550/ARXIV.2502.13923. URL https://doi.org/10.48550/arXiv.2502.13923.

Yizhak Ben-Shabat, Xin Yu, Fatemeh Sadat Saleh, Dylan Campbell, Cristian Rodriguez Opazo, Hongdong Li, and Stephen Gould. The IKEA ASM dataset: Understanding people assembling furniture through actions, objects and pose. In *IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021*, pp. 846–858. IEEE, 2021. doi: 10.1109/WACV48630.2021.00089. URL https://doi.org/10.1109/WACV48630.2021.00089.

Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *CoRR*, abs/2409.16283, 2024. doi: 10.48550/ARXIV.2409.16283. URL https://doi.org/10.48550/arXiv.2409.16283.

Jake Bruce, Michael D. Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal M. P. Behbahani, Stephanie C. Y. Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott E. Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, Proceedings of Machine Learning Research, pp. 4603–4623. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/bruce24a.html.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xu Huang, Shu Jiang, Yuxin Jiang, Cheng Jing, Hongyang Li, Jialu Li, Chiming Liu, Yi Liu, Yuxiang Lu, Jianlan Luo, Ping Luo, Yao Mu, Yuehan Niu, Yixuan Pan, Jiangmiao Pang, Yu Qiao, Guanghui Ren, Cheng Ruan, Jiaqi Shan, Yongjian Shen, Chengshi Shi,

Mingkang Shi, Modi Shi, Chonghao Sima, Jianheng Song, Huijie Wang, Wenhao Wang, Dafeng Wei, Chengen Xie, Guo Xu, Junchi Yan, Cunbiao Yang, Lei Yang, Shukai Yang, Maoqing Yao, Jia Zeng, Chi Zhang, Qinglin Zhang, Bin Zhao, Chengyue Zhao, Jiaqi Zhao, and Jianchao Zhu. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *CoRR*, abs/2503.06669, 2025. doi: 10.48550/ARXIV.2503.06669. URL https://doi.org/10.48550/arXiv.2503.06669.

Delong Chen, Willy Chung, Yejin Bang, Ziwei Ji, and Pascale Fung. Worldprediction: A benchmark for high-level world modeling and long-horizon procedural planning. *CoRR*, abs/2506.04363, 2025a. doi: 10.48550/ARXIV.2506.04363. URL https://doi.org/10.48550/arXiv.2506.04363.

Delong Chen, Théo Moutakanni, Willy Chung, Yejin Bang, Ziwei Ji, Allen Bolourchi, and Pascale Fung. Planning with reasoning using vision language world model. *CoRR*, abs/2509.02722, 2025b. doi: 10.48550/ARXIV.2509.02722. URL https://doi.org/10.48550/arXiv.2509.02722.

Hanzhi Chen, Boyang Sun, Anran Zhang, Marc Pollefeys, and Stefan Leutenegger. Vidbot: Learning generalizable 3d actions from in-the-wild 2d human videos for zero-shot robotic manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pp. 27661–27672. Computer Vision Foundation / IEEE, 2025c. doi: 10.1109/CVPR52734.2025.02576. URL https://openaccess.thecvf.com/content/CVPR2025/html/Chen_VidBot_Learning_Generalizable_3D_Actions_from_In-the-Wild_2D_Human_Videos_CVPR_2025_paper.html.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 7310–7320. IEEE, 2024. doi: 10.1109/CVPR52733.2024.00698. URL https://doi.org/10.1109/CVPR52733.2024.00698.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *Int. J. Comput. Vis.*, 130(1):33–55, 2022. doi: 10.1007/S11263-021-01531-2. URL https://doi.org/10.1007/s11263-021-01531-2.

Google DeepMind. Gemini 3.1 pro: Model card. *https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-1-Pro-Model-Card.pdf*, 2026. URL https://storage.googleapis.com/deepmind-media/Model-Cards/Gemini-3-1-Pro-Model-Card.pdf.

Jiahua Dong, Qi Lyu, Baichen Liu, Xudong Wang, Wenqi Liang, Duzhen Zhang, Jiahang Tu, Hongliu Li, Hanbin Zhao, Henghui Ding, et al. Learning to model the world: A survey of world models in artificial intelligence. 2026. URL https://www.techrxiv.org/doi/pdf/10.36227/techrxiv.177274570.09578608/v1.

Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Häusser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 2758–2766. IEEE Computer Society, 2015. doi: 10.1109/ICCV.2015.316. URL https://doi.org/10.1109/ICCV.2015.316.

Fan Feng, Phillip Lippe, and Sara Magliacane. Learning interactive world model for object-centric reinforcement learning. *CoRR*, abs/2511.02225, 2025. doi: 10.48550/ARXIV.2511.02225. URL https://doi.org/10.48550/arXiv.2511.02225.

Quentin Garrido, Mahmoud Assran, Nicolas Ballas, Adrien Bardes, Laurent Najman, and Yann LeCun. Learning and leveraging world models in visual representation learning. *CoRR*, abs/2403.00504, 2024. doi: 10.48550/ARXIV.2403.00504. URL https://doi.org/10.48550/arXiv.2403.00504.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, Eugene Byrne, Zachary Chavis, Joya Chen, Feng Cheng, Fu-Jen Chu, Sean Crane, Avijit Dasgupta, Jing Dong, María Escobar, Cristhian Forigua, Abrham Gebreselasie, Sanjay Haresh, Jing Huang, Md Mohaiminul Islam, Suyog Dutt Jain, Rawal Khirodkar, Devansh Kukreja, Kevin J. Liang, Jia-Wei Liu, Sagnik Majumder, Yongsen Mao, Miguel Martin, Effrosyni Mavroudi, Tushar Nagarajan, Francesco Ragusa, Santhosh Kumar Ramakrishnan, Luigi Seminara, Arjun Somayazulu, Yale Song, Shan Su, Zihui Xue, Edward Zhang, Jinxu Zhang, Angela Castillo, Changan Chen, Xinzhu Fu, Ryosuke Furuta, Cristina González, Prince Gupta, Jiabo Hu, Yifei Huang, Yiming Huang, Weslie Khoo, Anush Kumar, Robert Kuo, Sach Lakhavani, Miao Liu, Mi Luo, Zhengyi Luo, Brighid Meredith, Austin Miller, Oluwatumininu Oguntola, Xiaqing Pan, Penny Peng, Shraman Pramanick, Merey Ramazanova, Fiona Ryan, Wei Shan, Kiran K. Somasundaram, Chenan Song, Audrey Southerland, Masatoshi Tateno, Huiyu Wang, Yuchen Wang, Takuma Yagi, Mingfei Yan, Xitong Yang, Zecheng Yu, Shengxin Cindy Zha, Chen Zhao, Ziwei Zhao, Zhifan Zhu, Jeff Zhuo, Pablo Arbeláez, Gedas Bertasius, Dima Damen, Jakob J. Engel, Giovanni Maria Farinella, Antonino Furnari, Bernard Ghanem, Judy Hoffman, C. V. Jawahar, Richard A. Newcombe, Hyun Soo Park, James M. Rehg, Yoichi Sato, Manolis Savva, Jianbo Shi, Mike Zheng Shou, and Michael Wray. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 19383–19400. IEEE, 2024. doi: 10.1109/CVPR52733.2024.01834. URL https://doi.org/10.1109/CVPR52733.2024.01834.

David Ha and Jürgen Schmidhuber. World models. *CoRR*, abs/1803.10122, 2018. URL http://arxiv.org/abs/1803.10122.

Danijar Hafner, Timothy P. Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=S1lOTC4tDS.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy P. Lillicrap. Mastering diverse control tasks through world models. *Nat.*, 640(8059):647–653, 2025. doi: 10.1038/S41586-025-08744-2. URL https://doi.org/10.1038/s41586-025-08744-2.

Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. Reasoning with language model is planning with world model. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pp. 8154–8173. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.EMNLP-MAIN.507. URL https://doi.org/10.18653/v1/2023.emnlp-main.507.

Mengkang Hu, Tianxing Chen, Yude Zou, Yuheng Lei, Qiguang Chen, Ming Li, Yao Mu, Hongyuan Zhang, Wenqi Shao, and Ping Luo. Text2world: Benchmarking large language models for symbolic world model generation. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Findings of ACL, pp. 26043–26066. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.findings-acl.1337/.

Jinbang Huang, Yixin Xiao, Zhanguang Zhang, Mark Coates, Jianye Hao, and Yingxue Zhang. One demo is all it takes: Planning domain derivation with llms from A single demonstration. *CoRR*, abs/2505.18382, 2025. doi: 10.48550/ARXIV.2505.18382. URL https://doi.org/10.48550/arXiv.2505.18382.

Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022. URL https://openreview.net/pdf?id=BZ5a1r-kVsf.

Jinhan Li, Yifeng Zhu, Yuqi Xie, Zhenyu Jiang, Mingyo Seo, Georgios Pavlakos, and Yuke Zhu. OKAMI: teaching humanoid robots manipulation skills through single video imitation. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, Proceedings of Machine Learning Research, pp. 299–317. PMLR, 2024. URL https://proceedings.mlr.press/v270/li25a.html.

Yixia Li, Hongru Wang, Jiahao Qiu, Zhenfei Yin, Dongdong Zhang, Cheng Qian, Zeping Li, Pony Ma, Guanhua Chen, Heng Ji, and Mengdi Wang. From word to world: Can large language models be implicit text-based world models? *CoRR*, abs/2512.18832, 2025. doi: 10.48550/ARXIV.2512.18832. URL https://doi.org/10.48550/arXiv.2512.18832.

Yiran Qin, Zhelun Shi, Jiwen Yu, Xijun Wang, Enshen Zhou, Lijun Li, Zhenfei Yin, Xihui Liu, Lu Sheng, Jing Shao, Lei Bai, and Ruimao Zhang. Worldsimbench: Towards video generation models as world simulators. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*, Proceedings of Machine Learning Research. PMLR / OpenReview.net, 2025. URL https://proceedings.mlr.press/v267/qin25f.html.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Yansong Tang, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. COIN: A large-scale dataset for comprehensive instructional video analysis. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 1207–1216. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00130. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Tang_COIN_A_Large-Scale_Dataset_for_Comprehensive_Instructional_Video_Analysis_CVPR_2019_paper.html.

Beichen Wang, Juexiao Zhang, Shuwen Dong, Irving Fang, and Chen Feng. VLM see, robot do: Human demo video to robot action plan via vision language model. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2025, Hangzhou, China, October 19-25, 2025*, pp. 17215–17222. IEEE, 2025a. doi: 10.1109/IROS60139.2025.11246682. URL https://doi.org/10.1109/IROS60139.2025.11246682.

Jianan Wang, Bin Li, Jingtao Qi, Xueying Wang, Fu Li, and Lihanxun Li. Besimulator: A large language model powered text-based behavior simulator. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 4736–4754. Association for Computational Linguistics, 2025b. doi: 10.18653/V1/2025.EMNLP-MAIN.237. URL https://doi.org/10.18653/v1/2025.emnlp-main.237.

Jianan Wang, Siyang Zhang, Bin Li, Juan Chen, Jingtao Qi, Zhuo Zhang, and Chen Qian. Logicenvgen: Task-logic driven generation of diverse simulated environments for embodied AI. *CoRR*, abs/2601.13556, 2026. doi: 10.48550/ARXIV.2601.13556. URL https://doi.org/10.48550/arXiv.2601.13556.

Ruoyao Wang, Graham Todd, Ziang Xiao, Xingdi Yuan, Marc-Alexandre Côté, Peter Clark, and Peter A. Jansen. Can language models serve as text-based world simulators? In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pp. 1–17. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-SHORT.1. URL https://doi.org/10.18653/v1/2024.acl-short.1.

Siyin Wang, Zhaoye Fei, Qinyuan Cheng, Shiduo Zhang, Panpan Cai, Jinlan Fu, and Xipeng Qiu. World modeling makes a better planner: Dual preference optimization for embodied task planning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025, pp. 21518–21537. Association for Computational Linguistics, 2025c. URL https://aclanthology.org/2025.acl-long.1044/.

Zizhao Wang, Kaixin Wang, Li Zhao, Peter Stone, and Jiang Bian. Dyn-o: Building structured world models with object-centric representations. *CoRR*, abs/2507.03298, 2025d. doi: 10.48550/ARXIV.2507.03298. URL https://doi.org/10.48550/arXiv.2507.03298.

Kaige Xie, Ian Yang, John Gunerli, and Mark O. Riedl. Making large language models into world models with precondition and effect knowledge. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pp. 7532–7545. Association for Computational Linguistics, 2025a. URL https://aclanthology.org/2025.coling-main.503/.

Senwei Xie, Hongyu Wang, Zhanqi Xiao, Yun-Ru Wang, and Xilin Chen. Robotic programmer: Video instructed policy code generation for robotic manipulation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2025, Hangzhou, China, October 19-25, 2025*, pp. 14923–14930. IEEE, 2025b. doi: 10.1109/IROS60139.2025.11247043. URL https://doi.org/10.1109/IROS60139.2025.11247043.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=sFyTZEqmUY.

Haoming Ye, Yunxiao Xiao, Cewu Lu, and Panpan Cai. Unidomain: Pretraining a unified PDDL domain from real-world demonstrations for generalizable robot task planning. *CoRR*, abs/2507.21545, 2025a. doi: 10.48550/ARXIV.2507.21545. URL https://doi.org/10.48550/arXiv.2507.21545.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Se June Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, Lars Liden, Kimin Lee, Jianfeng Gao, Luke Zettlemoyer, Dieter Fox, and Minjoon Seo. Latent action pretraining from videos. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025b. URL https://openreview.net/forum?id=VYOe2eBQeh.

Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David F. Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 3537–3545. Computer Vision Foundation / IEEE, 2019. doi: 10.1109/CVPR.2019.00365. URL http://openaccess.thecvf.com/content_CVPR_2019/html/Zhukov_Cross-Task_Weakly_Supervised_Learning_From_Instructional_Videos_CVPR_2019_paper.html.

A LIMITATIONS

First, VIDEAS establishes the trajectory simulation and semantics extraction within a text-based environment. We adopt this formulation because natural language effectively abstracts complex visual observations and highlights the high-level causal logic of embodied actions. Although this abstraction facilitates efficient world modeling, it inherently discretizes state representations and discards continuous physical constraints such as kinematics, collision geometry, and fluid dynamics. Consequently, the current framework functions primarily as a semantic logic engine rather than a fully grounded continuous physics simulator. Second, the data distillation pipeline relies on VLMs to extract action semantics. Although we employ the prior-guided trajectory simulation mechanism to rigorously validate the extracted knowledge, latent hallucinations from VLMs might still introduce subtle noise into the knowledge corpus.

B TRAINING DETAILS

For model training, we apply the Low-Rank Adaptation technique to update all linear projections within the attention mechanisms and feed-forward networks. While Qwen3.5-9B serves as a VLM, we perform pure text fine-tuning and keep its visual encoder frozen. The training process takes approximately 7 hours for the Llama-3.1-8B-Instruct model, 7.5 hours for the Qwen3-8B model and 16 hours for the Qwen3.5-9B model (on four NVIDIA A800 GPUs). Table 4 presents the essential training hyperparameter configurations in our experiments. Additionally, the temperatures of LLMs and VLMs are set to 0 during evaluation.

Table 4: Training hyperparameters.

Hyperparameter	**Value**
Training epochs	3
Learning rate	1×10^{-4}
Warmup ratio	0.03
Batch size	4
Gradient accumulation steps	1
LoRA rank	32
LoRA alpha	64
LoRA dropout	0.1
Precision	bfloat16
Maximum sequence length	4096

C EVALUATION DETAILS

Our evaluation benchmarks comprise BTSIMBENCH (Wang et al., 2025b), LogicEnvEval (Wang et al., 2026), and WorldPrediction-WM (Chen et al., 2025a). During the evaluation on BTSIMBENCH and LogicEnvEval, we explicitly provide two few-shot examples in the system prompt for the evaluated foundation models to enhance their instruction following capabilities.

C.1 BTSIMBENCH

For the BTSIMBENCH evaluation, we integrate various foundation models into the BeSimulator framework (Wang et al., 2025b). This integration allows us to assess the action semantic reasoning capabilities of the models by deploying them as the core predictive engines within the embodied simulated environment. This benchmark contains three categories of robot policies, and we adopt the delivery rate and the simulation accuracy across these three categories as evaluation metrics. Specifically, the delivery rate measures whether the model successfully delivers the final simulation result within a predefined iteration limit (5 times in our experiment settings). Common reasons for delivery failure include response timeouts and formatting errors. The simulation accuracy represents the ratio of policies that the model correctly evaluates for each specific category. It requires the

model to precisely capture execution preconditions and predict environment transitions during the simulation process.

C.2 LogicEnvEval

The LogicEnvEval benchmark is originally used to evaluate simulated environment generation. We utilize its task data and 100 agent policies spanning four distinct categories. Then, we evaluate the foundation models through robot behavior simulation like BTSIMBENCH.

C.3 WorldPrediction-WM

Regarding the WorldPrediction-WM benchmark, we encounter accessibility and permission constraints with several original source video datasets (COIN (Tang et al., 2019), CrossTask (Zhukov et al., 2019), and EgoExo4D (Grauman et al., 2024)). Consequently, we use an evaluation subset consisting of 637 accessible task samples. To ensure fair comparisons, we evaluate all foundation models exclusively on this exact subset. Table 5 details the precise distribution of these samples. Given the initial and final world states, this benchmark requires the evaluated models to deduce the correct action responsible for the environment transition from a set of counterfactual distractors. For VLMs, these states are directly provided as visual inputs comprising the start and end video frames. Because VIDEAS-WM-9B retains the frozen visual encoder from its base model, it directly processes the visual frames and is thereby categorized under the VLMs group during evaluation. For the Socratic LLMs such as DeepSeek-V4-Pro, we follow the official setting of WorldPrediction-WM, which decouples perception and reasoning into two separate stages. Specifically, the visual inputs are first translated into textual descriptions through a VLM, and the LLM under evaluation is then prompted with these captions together with the structured task explanations and candidate actions. The LLM subsequently performs textual reasoning to identify the correct action. To obtain the textual descriptions, we follow the official setting and utilize Qwen2.5-VL-72B (Bai et al., 2025).

Table 5: Distribution of accessible task samples in WorldPrediction-WM.

Dataset	**Number of Samples**
COIN	195
CrossTask	90
IKEA-ASM	159
EPIC-KITCHENS-100	193
Total	637

D Efficiency Analysis

We conduct an efficiency analysis on the BTSIMBENCH benchmark to evaluate the inference latency of various foundation models. Specifically, we measure the average time required to simulate a single robot policy in minutes. To intuitively illustrate the performance and efficiency trade-off, we present a Pareto frontier scatter plot in Figure 5, where the horizontal axis denotes the simulation time cost and the vertical axis represents the average simulation accuracy. Models situated in the top-left region of this plot achieve the optimal balance between high reasoning capability and low computational overhead. Notably, our VIDEAS-WM consistently occupies this optimal region. Despite smaller parameter scales, VIDEAS-WM not only surpasses larger models such as Gemma4-31B in embodied action semantic reasoning but also exhibits significantly reduced inference latency.

E Text-based Environment

We provide a concrete example to illustrate the structure of text-based simulated environments, as shown in Figure 6. It presents a typical kitchen scenario where the agent receives instructions to interact with household appliances and retrieve target objects. The environment state is formalized into a hierarchical JSON format, capturing room layouts, physical entities, spatial relations,

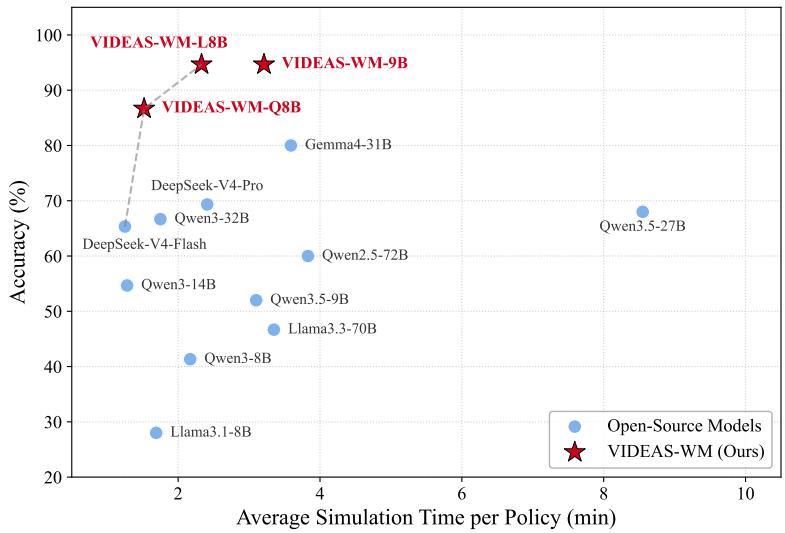


Figure 5: Pareto frontier of accuracy versus average simulation time per robot policy. VIDEAS-WM establishes the optimal balance between reasoning capability and computational efficiency.

and non-spatial relations. This structured context provides the VLMs with a fully observable and deterministic physical world to determine action preconditions and track state transitions.

F Prompt

This section details the prompts used in the trajectory simulation mechanism. The main text states that this mechanism models the execution of atomic actions through three phases to provide a concise high-level overview. The practical implementation operates through four distinct computational stages. We decouple the final transfer phase into a capture stage and a subsequent transfer stage. This decoupling reduces the reasoning burden on VLMs and ensures higher fidelity in predicting complex environmental dynamics. Using the Toast Bread episode as an example, we present the prompts and ideal VLM responses corresponding to the four stages of a specific action (Figure 7 - 10).

G Negative Trajectory Case

Figure 11 illustrates the synthesis process of a negative trajectory, exemplified by a representative episode of *retrieving clothes from a washing machine*.

H Case Study

In Figure 12, we present a case study illustrating how VIDEAS-WM-9B simulates a predefined robot policy for the BoilWater task from the BTSIMBENCH benchmark.



Task Name: Open the fridge to get food.
Action Trajectory: (1) Open the refrigerator door with the left arm. → (2) Pick up the cola from the fridge with the left arm. → (3) Place the cola on the table. → (4) Close the refrigerator door with the right arm.

Text-Based Embodied Environment (Abbreviated JSON Structure)


```
{
    "rooms": {
        "kitchen": {
            "vertices": [[0, 0], [4, 0], [4, 5], [0, 5]],
            "floor": "white ceramic tile, glossy",
            "wall": "light beige drywall, smooth"
        }
    },
    "...": "...",
    "agents": {
        "service_robot": {
            "description": "A humanoid service robot with two arms for manipulation",
            "size": 1.2,
            "gripper_contact_range": 1.2,
            "located_room": "kitchen",
            "position": [3.62, 3.87, 0.6]
        }
    },
    "spatial_relation": {
        "cola_bottle_refrigerator_1": {
            "source": "cola_bottle",
            "type": "in",
            "target": "refrigerator",
            "parameter": null
        },
        "refrigerator_north_1": {
            "source": "refrigerator",
            "type": "edge",
            "target": "north",
            "parameter": null
        },
        "...": "..."
    },
    "non_spatial_relation": {},
    "objects": {
        "refrigerator": {
            "description": "A stainless steel household refrigerator with a single
     door",
            "size": [0.7, 0.8, 1.8],
            "located_room": "kitchen",
            "open_close_state": "closed",
            "content": ["cola_bottle"],
            "power_state": "on",
            "position": [3.62, 4.56, 0.9],
            "direction": 180
        },
        "cola_bottle": {
            "description": "A plastic bottle of cola with a screw cap",
            "size": [0.08, 0.08, 0.2],
            "located_room": "kitchen",
            "content": [["cola"]],
            "position": [3.5, 4.5, 1.0],
            "direction": 0
        },
        "kitchen_table": {
            "description": "A rectangular wooden table with a smooth surface",
            "size": [1.0, 0.8, 0.75],
            "located_room": "kitchen",
            "position": [3.56, 3.0, 0.37],
            "direction": 270
        },
        "...": "..."
    }
}
```


Figure 6: An illustrative example of the text-based environment. The complete JSON structure encompasses comprehensive fine-grained properties, which are abbreviated here for brevity.

Prompt

You are a world model that can recognize and understand various scenes in the real world well, and can determine whether the action could be executed in the current scene.

Task Description

Based on your understanding of the textual current world states, the semantics of the given robot action, and the environmental observations before and after action execution (start frame and end frame), your task is to think the preconditions of the robot action. And you need to give your reason process.

Output Rules

1. Your output must be a dictionary. Just three keys are included: 'thought', 'corestates', and 'corestates_successtag'. Please do not output irrelevant content.

2. In the 'thought' key, you should first summarize the conditions that need to be met and the corresponding boolean value for the action to be executed successfully. Then, identify which states in the textual current world states are crucial for influencing each condition. Boolean value is true, indicating that these conditions should be met for the action execution; Boolean value is false, indicating that these conditions should not be met for the action execution.

3. In the 'corestates' key, the value is a dictionary. The dictionary includes all preconditions that affect the execution of the action. The keys of the dictionary should be expressed as complete question sentences, representing each precondition. The corresponding values should be the core states from the textual current world states that are necessary to check each precondition. The state names should be represented as A-B-C. Keys from different levels are connected with hyphen. Each condition corresponds to several states in a list.

4. In the 'corestates_successtag' key, it shows the boolean value that each precondition in the 'corestates' dictionary should return for the action to be executed successfully. Ensure that information in 'corestates_successtag' should be consistent with the meaning in 'corestates'.

5. You don't need to consider whether the robot's grippers are free.

6. The response should be output in the JSON format.

Input

Current States: [Text-Based Environment]

Robot Action: Insert the whole wheat toast held by the left arm into the slot behind white toaster.

Environmental Observations: start frame: [Start Frame] end frame: [End Frame]

Expected Output

```
{
    "thought": "...",
    "corestates_successtag": {
        "whether_robot_hold_toast_slice_2?": true,
        "is_white_toaster_back_slot_empty?": true,
        "is_white_toaster_inside_robot_gripper_contact?": true
    },
    "corestates": {
        "whether_robot_hold_toast_slice_2?": [
            "non_spatial_relation-
    toasting_robot_toast_slice_2_19704749-type"
        ],
        "is_white_toaster_back_slot_empty?": [
            "objects-white_toaster-back_slot_state"
        ],
        "is_white_toaster_inside_robot_gripper_contact?": [
            "agents-toasting_robot-position",
            "objects-white_toaster-position",
            "agents-toasting_robot-gripper_contact_range"
        ]
    }
}
```

Figure 7: The prompt structure and expected output for the ***consider*** stage.

Prompt

You are a world model that can recognize and understand various scenes in the real world well.

Task Description

Based on your understanding of the textual current world state, your task is to determine whether the provided condition is met based on the current states and the core states, and gives your reason process.

Output Rules

1. Your output must be a dictionary. Just Two keys are included: ”thought” and ”result”. Please do not output irrelevant content.

2. In the 'thought' key, you should give your reasoning process about whether the condition is true based on the current world states and provided core world states.

3. In the 'result' key, you need to output the judge result on whether the condition is true or false. 'True' indicates that the condition is true in the current world state. 'False' indicates that the condition is false in the current world state.

4. The response should be output in the JSON format.

Input

Current States: [Text-Based Environment]

Condition: name: is_white_toaster_back_slot_empty? core states: ['objects-white_toaster-back_slot_state']

Expected Output

```
{
    "thought": "...",
    "result": true
}
```

Figure 8: The prompt structure and expected output for the ***decide*** stage.

Prompt

You are a world model that can recognize and understand various scenes in the real world well, and can predict the future world states if robot executes an action in the current world states.

Task Description

Based on your understanding of the textual current world states, the semantics of the given robot action, and the environmental observations before and after action execution (start frame and end frame), your task is to think all world states that need to be transferred after the action is executed. Based on the start frame and end frame, you need to compare the environment changes before and after the action execution to enhance reasoning. And please give your reason process.

Output Rules

1. Your output must be a dictionary. Just Two keys are included: "thought" and "states_transferred". Please do not output irrelevant content.

2. For "thought" key, please output your thought and reason process about the current situation and which states needed to be transferred.

3. For "states_transferred" key, its value is a dictionary which includes three keys: "insert", "delete" and "update". They respectively represent the relations to be added, the relations to be deleted and the states to be updated.

4. For "insert" key, its value is a dictionary including two keys: "spatial_relation" and "non_spatial_relation". Their value is a nested list where each sub-list is a string list of length 3, respectively the source party, the relation type and the object party.

5. For "delete" key, its value is a list which indicates the relations to be deleted. Each relation should be represented as A-B. Keys from different levels are connected with hyphens.

6. For "update" key, its value is a list which indicates the states to be updated. Each state should be represented as A-B-C. Keys from different levels are connected with hyphens.

7. The response should be output in the JSON format.

Input

Current States: [Text-Based Environment]

Robot Action: Insert the whole wheat toast held by the left arm into the slot behind white toaster.

Environmental Observations: start frame: [Start Frame] end frame: [End Frame]

Expected Output

```
{
    "thought": "...",
    "states_transferred": {
        "insert": {
            "spatial_relation": [
                [
                    "toast_slice_2",
                    "in",
                    "white_toaster"
                ]
            ],
            "non_spatial_relation": []
        },
        "delete": [
            "non_spatial_relation-
    toasting_robot_toast_slice_2_19704749"
        ],
        "update": [
            "objects-toast_slice_2-position",
            "objects-white_toaster-back_slot_state"
        ]
    }
}
```

Figure 9: The prompt structure and expected output for the ***capture*** stage.

Prompt

You are a world model that can recognize and understand various scenes in the real world well, and can predict the future world states if robot executes an action in the current world states.

Task Description

Based on your understanding of the textual current world state, and the semantics of the given action, your task is to simulate the execution of the given action and predict the transition of the designated world state after the action happens. You need to follow real-world physical laws.

Output Rules

1. You need to output the new value of the designated state based on this action's semantic description and the information in *Thought* & *Changed States*.

2. The response should be output in the JSON format.

Input

Current States: [Text-Based Environment]

Robot Action: Insert the whole wheat toast held by the left arm into the slot behind white toaster.

Thought: [Thought Process of the capture phase]

Changed States: ['objects-toast_slice_2-position']

States That Need To Be Transferred: objects-white_toaster-back_slot_state

---

Expected Output

```
{
    "objects": {
        "white_toaster": {
            "back_slot_state": "occupied"
        }
    }
}
```

Figure 10: The prompt structure and expected output for the ***transfer*** stage.



Task Name: Remove clothes from the washing machine
Action Trajectory: (1) Open the washing machine door with both arms. → (2) Move forward to approach the washing machine. → (3) Retrieve shorts from the washing machine with the right arm. → (4) Place the shorts into the laundry basket. → (5) Retrieve a short sleeve from the washing machine. → (6) Place the short sleeve into the laundry basket. → (7) Step back a bit away from the washing machine. → (8) Close the washing machine door.

Dependency Records (Abbreviated JSON Structure)


```
{
    "778774": {
        "task_name": "Remove clothes from the washing machine",
        "actions": [
            {"index": 0, "name": "0_Open", "skill": "Open", "text": "Open
    the washing machine door with both arms."},
            {"index": 1, "name": "1_Move", "skill": "Move", "text": "Move
    forward to approach the washing machine."},
            {"index": 2, "name": "2_Pick", "skill": "Pick", "text": "
    Retrieve a shorts from the washing machine with the right arm."},
            ...
        ],
        "dependencies": [
            {
                "action_index": 2,
                "action_name": "2_Pick",
                "depends_on": [0],
                "dependency_type": "object_state",
                "precondition_nl": "washing machine door is open",
                "mutation_viable": true
            },
            {
                "action_index": 3,
                "action_name": "3_Place",
                "depends_on": [2],
                "dependency_type": "hold_object",
                "precondition_nl": "robot is holding a shorts",
                "mutation_viable": true
            },
            {
                "action_index": 4,
                "action_name": "4_Pick",
                "depends_on": [0],
                "dependency_type": "object_state",
                "precondition_nl": "washing machine door is open",
                "mutation_viable": true
            },
            "...": "..."
        ]
    }
}
```


Mutated Negative Trajectory: (1) Move forward to approach the washing machine. → (2) Retrieve a shorts from the washing machine with the right arm. → (3) Place the shorts into the laundry basket. → (4) Retrieve a short sleeve from the washing machine. → (5) Place the short sleeve into the laundry basket. → (6) Step back a bit away from the washing machine. → (7) Close the washing machine door.

Figure 11: An illustrative example of negative trajectories. By removing the initial *Open* action, the subsequent *Pick* action is deliberately designed to fail due to unmet physical preconditions.

Task Name: BoilWater
Task Description: The robot task is to boil water. The behavior logic of a robot should be as follows. The kettle is initially inside a cabinet and is filled with water. The stove and cabinet are all located in the kitchen. The robot needs to move the kettle from the cabinet to the stove, heat the water, and wait for that the water in the kettle is boiled. The goal is to get boiled water.

Action Sequence	Predicted Preconditions & Effects (JSON Structure)

(1) **Open_cabinet**: The robot grips the cabinet door handle, pulls it open smoothly, ensuring a safe and controlled motion.	Preconditions:
	`{ "is_cabinet_closed?": true, "is_cabinet_inside_robot_gripper_contact?": true }`
	Effects:
	`{ "states_transferred": { "insert": { "spatial_relation": [], "non_spatial_relation": [] }, "delete": [], "update": [ "objects-cabinet-open_close_state" ] } }`

(2) **Take_out_kettle**: The robot extends its arm, grasps the kettle, and takes it out from the cabinet.	Preconditions:
	`{ "is_cabinet_open?": true, "is_kettle_in_cabinet?": true, "is_kettle_inside_robot_gripper_contact?": true }`
	Effects:
	`{ "states_transferred": { "insert": { "spatial_relation": [], "non_spatial_relation": [ ["kitchen_robot", "hold", "kettle"] ] }, "delete": [ "spatial_relation-kettle_cabinet_1" ], "update": [ "objects-kettle-position" ] } }`

(3) **Move_to_stove**: The robot moves to the location of the stove.	Preconditions:
	`{}`
	Effects:
	`{ "states_transferred": { "insert": { "spatial_relation": [], "non_spatial_relation": [] }, "delete": [], "update": [ "agents-kitchen_robot-position", "objects-kettle-position" ] } }`

Figure 12: Case study illustrating the simulation of a robot policy from the BTSIMBENCH benchmark using VIDEAS-WM-9B. For each action within the sequence, the predicted preconditions and effects are formalized in JSON format.

Action Sequence	Predicted Preconditions & Effects (JSON Structure)

(4) **Place_kettle_on_stove**: The robot places the kettle on the stove smoothly.	Preconditions:
	`{ "is_stove_inside_robot_gripper_contact?": true, "whether_robot_hold_kettle?": true }`
	Effects:
	`{ "states_transferred": { "insert": { "spatial_relation": [ ["kettle", "on_top_of", "stove"] ], "non_spatial_relation": [] }, "delete": [ "non_spatial_relation- kitchen_robot_kettle_c7af0617" ], "update": [ "objects-kettle-position" ] } }`

(5) **Turn_on_stove**: The robot turns on the stove switch to start heating.	Preconditions:
	`{ "is_stove_inside_robot_gripper_contact?": true, "is_stove_off?": true }`
	Effects:
	`{ "states_transferred": { "insert": { "spatial_relation": [], "non_spatial_relation": [] }, "delete": [], "update": [ "objects-stove-power_state" ] } }`

(6) **Wait_water_boiled**: The robot waits for the stove to heat the kettle until the water in the kettle reaches boiling point.	Preconditions:
	`{ "is_kettle_on_stove?": true, "kettle_has_water?": true, "kettle_is_closed?": true, "kettle_is_not_boiled?": true, "stove_is_on?": true }`
	Effects:
	`{ "states_transferred": { "insert": { "spatial_relation": [], "non_spatial_relation": [] }, "delete": [], "update": [ "objects-kettle-temperature", "objects-kettle-boiling_state" ] } }`

Figure 12: Case study illustrating the simulation of a robot policy from the BTSIMBENCH benchmark using VIDEAS-WM-9B (Continued).