Being-H0.7: A Latent World-Action Model from Egocentric Videos

BeingBeyond Team

https://research.beingbeyond.com/being-h07

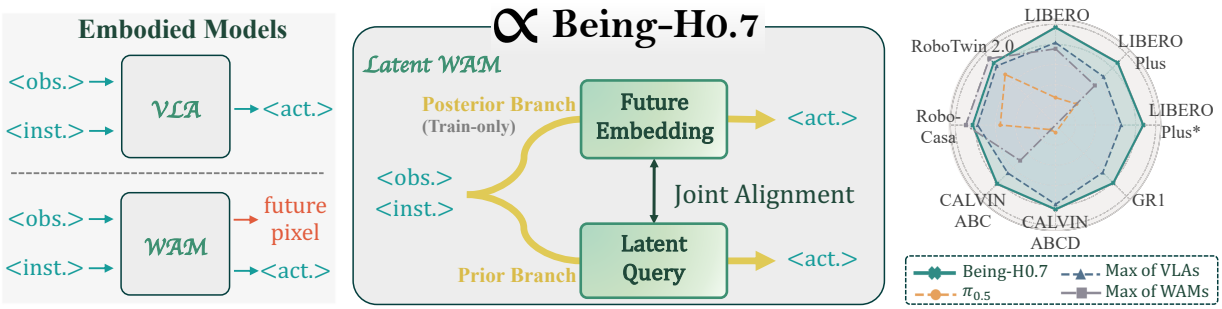


Figure 1: **Being-H0.7 at a glance.** We build a Latent World-Action Model that differs from VLAs and WAMs. A latent reasoning space is introduced via a set of latent queries in the prior branch, and is further endowed with world modeling by the joint alignment with a future-aware posterior branch. Pretrained on large-scale egocentric videos, Being-H0.7 achieves strong performance across diverse robot tasks while remaining efficient in both training and inference stages.

Abstract

We present Being-H0.7, a *latent world-action model* that generates actions by modeling world dynamics in the latent space. Visual-Language-Action models (VLA) have demonstrated strong capabilities across diverse robotic tasks by directly mapping observations to actions. However, limited and sparse action supervision often leads to behavior collapse, preventing these models from learning physically grounded and diverse action representations. Recent works instead introduce world-action models (WAMs) built upon video generation, which attempt to capture future evolution through dense pixel prediction, leveraging large-scale video pretraining to improve generalization. Yet, they require future trajectory rollouts at inference time, resulting in substantial latency, and may suffer from imperfect predicted pixels. Our choice is to bridge them via latent world modeling, introducing a compact latent reasoning space between perception and action. We instantiate this with a small set of learnable latent queries that support a compact reasoning interface carrying future-relevant information before action generation. To shape the reasoning space efficiently, we use a dual-branch design that aligns the queries with future-aware embeddings in the latent space. In experiments, Being-H0.7 delivers state-of-the-art overall performance across six simulation benchmarks and completes challenging real-world tasks that require dynamic prediction and motion reasoning.

Date: Apr 14, 2026

1 Introduction

Visual-Language-Action models (VLAs) [1–7] have achieved remarkable progress across a wide range of robotic manipulation tasks, demonstrating that multimodal perception can be effectively translated into action generation. However, despite the semantically rich nature of visual and language observations, both task specifications and action data remain sparse and limited in diversity. As a result, existing VLAs often collapse a wide spectrum of situations into a small set of recurring behaviors, relying on spurious correlations rather than learning physically grounded and compositional action representations.

In parallel, world modeling has emerged as a promising direction for enhancing action learning by incorporating future awareness. By modeling future trajectories or interaction outcomes, world models provide additional structure beyond immediate observations and, in principle, enable more robust long-horizon reasoning. Recent works [8–10] explore leveraging large-scale robot data together with "no-action" internet videos for this purpose. However, these approaches typically operate in pixel space and rely on iterative video generation to simulate future rollouts at inference time. This introduces substantial latency, and errors in predicted futures can propagate into action decoding, leading to failures in long-horizon or dynamically evolving scenarios.

To mitigate these issues, recent progress attempts to decouple future prediction from action generation, enabling low-latency control by optionally skipping explicit visual rollout [11]. Nevertheless, this paradigm remains fundamentally dependent on video generation models, inheriting their limitations in efficiency, stability, and robustness. A critical bottleneck is their prohibitive computational cost: for example, Cosmos Policy [9] requires over 3,000 H100 GPU hours to train on LIBERO, whereas a representative VLA such as Being-H0.5 [7] converges within approximately 50 GPU hours, yielding a $60\times$ efficiency gap. Such costs scale poorly with data size, making it impractical to extend from thousands of robot episodes to internet-scale video corpora.

Given these trends, a question arises whether instruction understanding, world modeling, and action generation can be unified within a single framework in an efficient way. Future world evolution is clearly useful for action generation, but modeling it directly in observation space is often expensive, noisy, and only weakly aligned with downstream control. We argue that the key is to bridge direct action prediction and world modeling through a shared latent space. Such a space should be abstract enough to capture task-relevant semantics, compact enough to filter out pixel-level redundancy, and action-grounded enough to support dense control. From this perspective, we introduce a set of latent queries between multimodal context and action generation to support such a reasoning space. In the model propagation, the queries serve as a structured interface where the model first forms an intermediate latent state from context, and downstream action generation is conditioned on this state. We further seek to shape this latent reasoning space into a world-modeling substrate, so that future-relevant information can be organized in the latent space rather than predicted explicitly in observation space.

To this end, we further introduce a future-informed dual-branch design to explicitly shape the latent reasoning space toward world modeling. During training, a *prior branch* reasons from the current multimodal context alone, while an auxiliary *posterior branch* is additionally exposed to future observations. Concretely, we replace the queries with future embeddings in the posterior branch and leave the rest identical. The supervision in the posterior branch will push the model to capture what kind of future-relevant information is actually useful for action generation. By jointly aligning the two branches in the latent space, the model encourages the prior branch to infer what matters for task execution from the context. To further stabilize learning, we regularize the latent states with norm and rank constraints, preventing both magnitude shrinkage and directional collapse. This design allows us to unify instruction understanding, world modeling, and action generation in a single efficient framework. We retain only the prior branch at inference time, but the future-informed alignment has already shaped the latent reasoning space with world-action modeling. As a result, such a paradigm inherits the action efficiency of direct VLA-style policies, while gaining a stronger ability to capture future-relevant structure. We therefore view it as a **Latent World-Action Model**: the world is modeled not through explicit future frames, but through an action-oriented latent space that reasons about how interaction may unfold before producing control.

Based on this design, we present **Being-H0.7**, a latent world-action model pretrained on large-scale egocentric videos. Following our prior Being-H series, we also continue to scale up pretraining on large-scale human

videos, exposing the model to a breadth of tasks, environments, and interaction patterns that would be difficult to obtain from simulation or real-robot data alone. From Being-H0 to Being-H0.5, we were, to the best of our knowledge, the first to scale human-centric pretraining from 1,000 hours to over 10,000 hours, demonstrating the feasibility and effectiveness of this paradigm. In this work, we extend this direction by curating a massive egocentric dataset of 200,000 hours, approximately 15× larger than Being-H0.5. Benefit from both the training paradigm and the scaled data, Being-H0.7 delivers state-of-the-art overall performance across six simulation benchmarks and completes challenging real-world tasks that require dynamic prediction and motion reasoning.

Table 1: Comparison of Being-H model variants.

	Being-H0	**Being-H0.5**	**Being-H0.7**
Model Paradigm	VLA	VLA	Latent WAM
Action Head	MLP	Flow Matching	Flow Matching
Temporal Context	None	None	1-second horizon
Pretrained Embodiments	N/A	30	32
Egocentric Human Data (hrs)	1,000	16,000	200,000
Robot Demonstrations (hrs)	No	14,000	15,000
General VL Knowledge	No	Yes	No
Dynamic Environments	✗	✗	✓
Liquid Manipulation	✗	✗	✓
Deformable Objects	✗	✓	✓
Tool Use	✗	✓	✓

2 Related Work

Vision-Language-Action Models. Recent advances in robotic manipulation [12–15] have shifted from narrow, single-task specialists toward generalist models trained on diverse, large-scale datasets. Among them, Vision-Language-Action models (VLAs) [1–3, 16, 17] adapt pretrained vision-language models (VLMs) [18–24] for robotic control, and are effective at predicting actions directly from current observations. A key line of progress lies in action-head design: early methods rely on autoregressive tokenized actions [3, 25], while recent approaches increasingly adopt diffusion-based [26, 27] generators [4, 5, 16, 28–30], which improve efficiency and precision for complex control [31–33]. To better support high-level reasoning, some works further introduce textual planning or structured intermediate representations, including Chain-of-Thought (CoT [34]) planning [35–37] and spatial abstractions [38] such as bounding boxes [39], dense correspondence fields [40], 3D points [41], or trajectory traces [42]. However, these methods still primarily infer actions from the current observation, without explicitly modeling how the world may evolve through interaction.

World-Action Models. Recent work has increasingly explored video generation and world modeling as a foundation for robot control, motivated by the observation that video models capture temporal dynamics and plausible future evolution that are largely absent from static vision-language pretraining. One line of work uses video models primarily as predictive representation learners or transferable world priors, followed by separate action decoding modules [43–46]. A second line moves toward tighter coupling by jointly modeling future video and action within a unified architecture [47–49], showing that jointly predicting visual futures and action sequences can improve generalization and data efficiency. More recent works build on increasingly stronger pretrained video foundation models [50–52], and further push toward unified world-action models that support closed-loop control, causal rollout, or planning over predicted futures [53, 54]. DreamZero [8], Cosmos Policy [9], and LingBot-VA [10] exemplify this trend, showing that stronger video priors can be carried into embodied policies to improve generalization and embodiment transfer. Fast-WAM [11] shows that retaining video co-training during training while removing test-time future generation can preserve strong action performance with substantially lower latency. Additionally, a related but distinct line [55–58], inspired

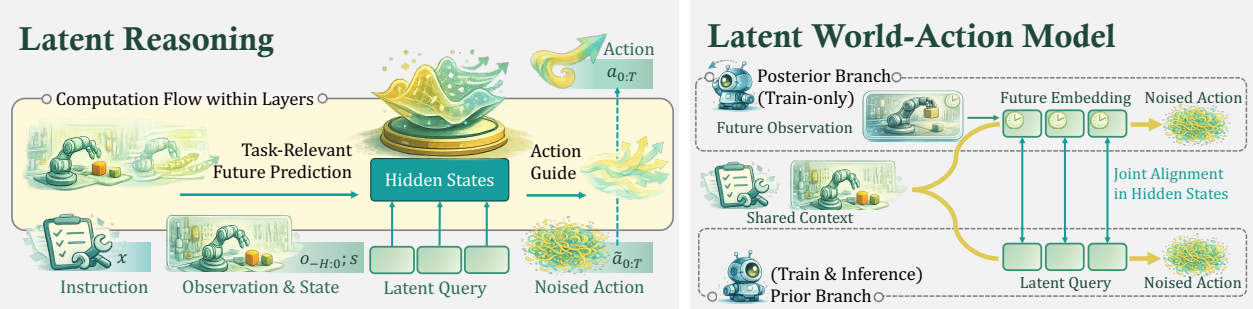


Figure 2: **Latent reasoning and latent world-action model. Left:** Learnable latent queries are inserted to form a latent reasoning space that progressively organizes intermediate hidden states and guides action generation through propagation. **Right:** Through joint alignment between the dual-branch design, the model learns to reason with future information at inference time, turning into a latent world-action model.

by JEPA [59], seeks to model future evolution through joint predictive representations. Our work is most closely related to this emerging world-action modeling line, but differs in that we do not rely on explicit future video rollout or stop at future-aware representation learning. Instead, we use future information to shape a deployable *explicit latent reasoning* process that directly participates in action generation.

3 Method

3.1 Latent Reasoning: At the Crossroads of VLA and World-Action Model

An effective embodied model should not only react to the instant context but also truly understand how the interaction may unfold. The progress in VLAs and video-generative world-action models highlights these two complementary aspects. Standard VLA models excel at directly mapping current observations to actions, but they do not explicitly model how the world may evolve under interaction. In contrast, video-generative world models attempt to capture such future evolution through dense pixel prediction, but this is both computationally expensive and poorly matched to the abstraction of physical dynamics. We argue that the key is not to choose between these two paradigms, but to connect them through a latent reasoning space: an explicit intermediate space where future-relevant, action-oriented information can be organized before generating low-level actions.

As illustrated in Fig. 2 (Left), we instantiate this idea by introducing a small set of learnable latent queries into the backbone, placing them between the multimodal context and the noised actions. Concretely, let x denote the instruction, $o_{-H:0}$ the observation context of horizon H, and s the state. We insert a set of latent queries $Q \in \mathbb{R}^{K\times d}$ before the action chunk, yielding the augmented sequence

$$S = \left[x;\, o_{-H:0};\, s;\, Q;\, a_{0:T}\right], \tag{1}$$

where K is the number of latent queries, d is the hidden dimension, and $a_{0:T}$ denotes an action chunk of length T. These latent queries define the latent reasoning space, which participates in the layer-by-layer Transformer propagation together with the instruction, observation, state, and action. Through repeated interaction across layers, they progressively integrate task-relevant information from the multimodal context, organize it into an action-oriented latent state, and in turn shape downstream action generation. In this way, the model is no longer forced to map abstract multimodal semantics directly into dense low-level actions. Instead, it can gradually form a compact intermediate reasoning state during forward propagation and use it to guide action prediction.

However, this formulation alone does not guarantee that future prediction will actually emerge within the latent reasoning process. When trained only with action supervision, the latent may instead collapse to a weak intermediate representation or encode only shallow cues sufficient for local action decoding. We therefore introduce in the next subsection a future-informed alignment mechanism to explicitly shape this latent reasoning as world modeling.

3.2 Latent World-Action Model: Joint Alignment with Future Information

While the latent reasoning space introduced in Sec. 3.1 provides an explicit substrate for intermediate reasoning, it does not by itself guarantee that the latent queries will organize meaningful future-relevant structure. To shape this latent reasoning space with future information while preserving a deployable inference pathway, we introduce a dual-branch training design, as illustrated in Fig. 2 (Right).

Dual-Branch Design. We construct two structurally matched branches that share the same context, backbone, and action generation pathway. The *prior branch* is the main deployable branch, where the action generation is conditioned only on the current instruction, observation context, state, and a set of learnable latent queries. In parallel, we introduce a training-only *posterior branch*, which has access to the future observations $\tilde{o}_{0:T}$. We replace the latent queries in the posterior branch with a compact set of future embeddings of the same shape, so that the two branches remain structurally aligned at the latent reasoning positions. Concretely, the future observations are first encoded by a frozen pretrained ViT, and then aggregated by a Perceiver resampler into K future embeddings,

$$z^{\text{post}} = E(\tilde{o}_{0:T}) \in \mathbb{R}^{K \times d}, \tag{2}$$

where E denotes the temporal visual encoder composed of the frozen ViT and the Perceiver resampler. Here, K matches the number of latent queries in the prior branch, and d is the hidden dimension. Under action supervision, the two branches naturally capture different views of reasoning for action generation. The prior branch encourages the model to first organize a latent reasoning state from the current context and then generate actions from this latent reasoning state. In contrast, the posterior branch is to reveal which future information is truly useful for action decision-making. By replacing the latent queries with future embeddings, it provides a future-informed version of the reasoning space and highlights the part of future evolution that should matter for downstream action generation.

Joint Alignment. We then introduce joint alignment on the hidden states of the two branches at the latent reasoning positions, so that these two views explicitly meet in the same latent space. Formally, let h_ℓ^{prior} and h_ℓ^{post} denote the aligned hidden states at layer ℓ from the prior and posterior branches, respectively. We apply the following alignment loss:

$$\mathcal{L}_{\text{align}} = \frac{1}{L} \sum_{\ell=1}^{L} \left\| h_\ell^{\text{prior}} - h_\ell^{\text{post}} \right\|_2^2, \tag{3}$$

where L is the number of aligned layers. Through this future-informed joint alignment, the latent reasoning space is no longer merely an intermediate carrier for action decoding. Instead, it is explicitly shaped to encode future-relevant, action-oriented structure. In this sense, the resulting model can be viewed as a *latent world-action model*: future information is introduced only during training, yet its effect is realized through the latent reasoning pathway that remains fully executable at inference time.

3.3 Efficient Dual-Branch Implementation

We implement the latent world-action model in a structurally-simple and training-efficient way, as illustrated in Fig. 3. Instead of running two fully separate forward passes, we pack the prior and posterior branches into a single sequence with a Mixture-of-Transformers (MoT) [60] structure. The two branches share the same current context tokens, while their branch-specific tokens occupy different latent reasoning positions: the prior branch uses learnable latent queries before actions, and the posterior branch uses future embeddings of the same shape.

To preserve the intended dual-branch structure within one packed sequence, we apply a dual-branch attention mask. Shared context tokens are visible to both branches, while the prior and posterior branch tokens are isolated from each other except through the explicitly aligned latent reasoning states. In addition, the positional IDs of the two branches are kept identical at corresponding token positions, so that the prior latent queries and posterior future embeddings remain structurally matched throughout the Transformer layers. This design allows the model to realize the dual-branch latent world-action formulation efficiently within a single backbone forward pass.

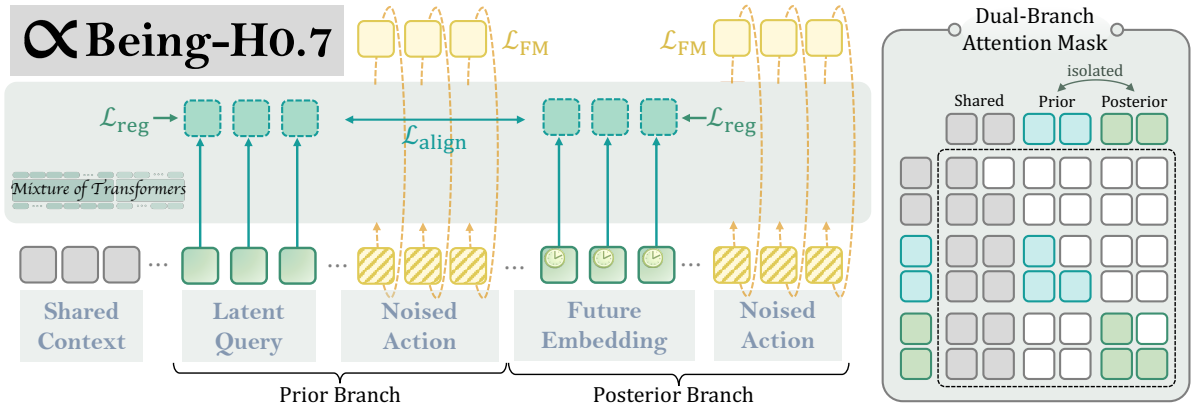


Figure 3: **Being-H0.7 Architecture.** We pack the prior and posterior branches into a single MoT sequence with shared context, where the two branches are optimized simultaneously. The posterior branch replaces latent queries with future embeddings, and the two branches are coupled by hidden-state alignment and lightweight regularization. A dual-branch attention mask is applied to isolate prior and posterior branches while preserving access to the shared context for efficient training.

To train the two branches, we apply a flow-matching objective for action generation on both the prior and posterior noised-action streams. Let a denote the action label, $t \in [0,1]$ the flow time, and $\epsilon \sim \mathcal{N}(0,1)$ the sampled noise. We construct the interpolated action $\tilde{a}_t = t\,a + (1-t)\cdot\epsilon$, with target velocity $u_t = a - \tilde{a}$. Let $v_\theta^{\text{prior}}(a_t, c)$ and $v_\theta^{\text{post}}(a_t, c, z^{\text{post}})$ denote the predicted velocity fields of the prior and posterior branches, respectively, where $c = [x;\, o_{-H:0};\, s]$ is the shared current context. We calculate the flow-matching objective

$$\mathcal{L}_{\text{FM}}^{\text{prior}} = \left\| v_\theta^{\text{prior}}(a_t, c) - u_t \right\|_2^2, \quad \mathcal{L}_{\text{FM}}^{\text{post}} = \left\| v_\theta^{\text{post}}(a_t, c, z^{\text{post}}) - u_t \right\|_2^2, \tag{4}$$

and combine them as

$$\mathcal{L}_{\text{FM}} = \mathcal{L}_{\text{FM}}^{\text{prior}} + \mathcal{L}_{\text{FM}}^{\text{post}}. \tag{5}$$

Since hidden-state alignment can otherwise admit trivial collapse, we apply a lightweight regularization term to both branches. Unlike strong representation-learning regularizers such as SIGReg [61] or VICReg [62], which impose a heavily structured feature geometry, we slightly regularize the hidden states of the latent reasoning space from two aspects: their norm and their rank. For norm regularization, given a latent hidden state h, we use

$$\mathcal{R}_{\text{norm}}(h) = \left[\text{ReLU}(\tau - \|h\|_2)\right]^2, \tag{6}$$

where τ is a predefined norm threshold, to prevent the aligned states from collapsing toward vanishing magnitude. For rank regularization, let $H \in \mathbb{R}^{B\times n}$ denote the projection on a random n-dim subspace of a batch of latent hidden states from one branch at one aligned layer. We normalize each row of H to unit norm to obtain $\hat{H}$, compute the Gram matrix $G = \hat{H}\hat{H}^\top$, and let $\{\lambda_i\}_{i=1}^B$ be the eigenvalues of G with normalized spectrum $p_i = \lambda_i / \sum_{j=1}^B \lambda_j$. We then use

$$\mathcal{R}_{\text{rank}}(H) = \sum_{i=1}^{B} p_i \log p_i, \tag{7}$$

which discourages directional collapse of the latent reasoning space. Applying both regularizers to the latent reasoning states from both branches and all aligned layers, we use a comprehensive regularization term,

$$\mathcal{L}_{\text{reg}} = w_{\text{norm}}\mathcal{R}_{\text{norm}} + w_{\text{rank}}\mathcal{R}_{\text{rank}}. \tag{8}$$

In practice, we pretrain the model on mixed human and robot manipulation data following the same unified format as UniHand 2.0. This provides a shared sequence structure for cross-embodiment action learning and

Table 2: Benchmark comparison on multiple embodied manipulation tasks. CALVIN denotes "ABCD → D" and CALVIN* denotes "ABC → D", LIBERO-plus* denotes finetuning with LIBERO-plus dataset

Model	Size	LIBERO	LIBERO-plus	LIBERO-plus*	RoboCasa-50	GR1	CALVIN	CALVIN*	Robotwin2
# VLA									
π0 [4]	3B	94.4	53.6	-	42.4	-	-	3.92	65.9/58.4
π0-FAST[63]	3B	85.5	61.6	-	-	-	-	-	-
X-VLA [64]	0.9B	-	-	-	-	-	4.43	-	72.9/72.8
UniVLA [65]	8B	95.5	-	-	-	-	4.63	4.41	-
gr00t-N1.6 [5]	3B	93.9	-	-	36.0	47.6	4.60	4.24	-
π0.5 [31]	3B	96.9	77.4	-	41.4	-	4.06	4.13	82.7/76.8
starVLA [66]	4B	96.5	77.0	-	-	48.8	-	-	88.2/88.3
MINT-4B [67]	4B	98.7	80.1	**84.1**	-	-	4.57	-	-
ABot-M0 [68]	4B	98.6	**80.5**	-	58.3	-	-	-	81.2/80.4
LingBot-VLA [69]	4B	-	-	-	-	-	-	-	86.5/85.3
Being-H0.5 [7]	2B	98.9	78.5	83.1	53.5	-	4.63	4.48	-
# World Model									
UWM [48]	-	79.0	-	-	48.2	-	-	-	-
UVA [47]	-	-	-	-	50.0	-	-	-	-
VPP [43]	1.5B	-	-	-	-	-	-	4.33	-
DreamVLA [70]	-	92.6	-	-	-	-	-	4.44	-
JEPA-VLA [71]	-	96.4	25.6	-	-	-	-	-	73.5/-
VLA-JEPA [58]	-	96.1	79.5	-	-	-	-	-	-
LingBot-VA [10]	5B	98.5	-	-	-	-	-	-	**92.9/91.6**
Cosmos-Policy [9]	7B	98.5	-	-	**67.1**	-	-	-	-
Fast-WAM [11]	6B	97.6	-	-	-	-	-	-	**91.9/91.8**
Being-H0.7	3B	**99.2**	**82.1**	**84.8**	**62.1**	**49.2**	**4.67**	**4.48**	90.2/89.6

allows the latent world-action model to be trained on heterogeneous manipulation trajectories in a unified way. Although the proposed architecture is, in principle, also compatible with the text-generation tasks in the same data format, we do not use those tasks in the current stage and focus only on action generation. The final training objective is

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + w_{\text{align}}\mathcal{L}_{\text{align}} + \mathcal{L}_{\text{reg}}. \tag{9}$$

During the training, we use an observation horizon of $H = 4$, an action chunk length of $T = 20$, a latent query num of $K = 16$, an alignment weight of $w_{\text{align}} = 10^{-3}$, and regularization weights of $w_{\text{norm}} = w_{\text{rank}} = 10^{-4}$.

4 Experiments

4.1 Simulation

4.1.1 Experimental Setup

Across all evaluations, our policy strictly relies on **RGB-only** observations, with all images uniformly resized to 224×224. Also, unless otherwise specified, we train the models using packed sequences, which maintains an effective batch size of around 128 across different datasets. All optimization processes is conducted on a standard computation node with 4×A800 GPUs.

We evaluate the Being-H0.7 model on the following six widely-used simulation benchmarks:

- **LIBERO** [72]: LIBERO is a comprehensive benchmark designed to evaluate knowledge transfer and lifelong learning capabilities in tabletop manipulation. It consists of four distinct task suites (Goal, Object, Spatial, and Long). We follow [72, 73] and train our model on data from all four suites. For evaluation, we conduct 500 trials per suite and report the average success rate across all suites.
- **RoboCasa** [74]: RoboCasa provides a large-scale simulation framework focusing on everyday long-horizon household tasks. We evaluate on the 24 base manipulation tasks within diverse kitchen environments and adopt the challenging **Human-50** few-shot setting, utilizing 50 human demonstrations per task. Evaluation is conducted over 50 trials per task across held-out scenes, specifically testing the model's robustness to unseen object instances and novel kitchen styles.

- **GR1** [5]: GR1 is a bimanual manipulation benchmark featuring a GR-1 humanoid robot equipped with Fourier dexterous hands. It comprises 24 complex tabletop manipulation tasks that require fine-grained dexterity and coordination. We train our model using 1000 demonstrations per task. Evaluation is performed with 50 trials per task.
- **LIBERO-plus** [75]: LIBERO-plus is explicitly designed to systematically assess policy robustness and zero-shot generalization under a diverse set of controlled environmental perturbations. Following standard practice [75], we evaluate our model under two distinct training configurations: a baseline trained exclusively on the standard LIBERO dataset, and a variant fine-tuned on the augmented LIBERO-plus dataset.
- **RoboTwin 2.0** [76]: RoboTwin 2.0 is a comprehensive framework designed to benchmark robust bimanual robotic manipulation. To systematically assess and enhance sim-to-real transfer, the benchmark incorporates structured domain randomization along five axes: table-top clutter, varied lighting conditions, diverse background textures, tabletop height variations, and diverse language instructions. We train our model on 2,500 demonstrations from clean scenes (50 per task) and 25,000 from highly randomized scenes (500 per task). We evaluate the policy under two distinct settings: **Easy** (clean scenes) and **Hard** (domain-randomized scenes), with 100 rollouts per task.
- **CALVIN** [77]: CALVIN is a benchmark that specifically targets multi-task learning and long-horizon manipulation capabilities across four distinct environments (A, B, C, and D). Following the standard evaluation protocol, we assess our model on two data splits: ABCD→D (training across all environments and testing on seen environment D) and ABC→D (testing zero-shot generalization to the unseen environment D). The evaluation is rigorously performed over 1,000 unique instruction sequences, where each sequence requires the agent to execute 5 consecutive instructions. We report the average number of tasks completed per sequence.

4.1.2 Results

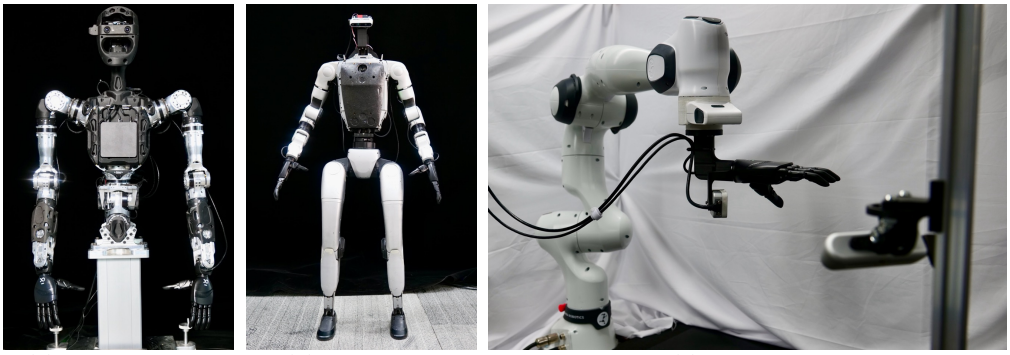
Across all six simulation benchmarks, Being-H0.7 achieves state-of-the-art overall performance, maintaining the highest average rank as detailed in Table 2. On **LIBERO**, Being-H0.7 reaches a **99.2%** average success rate, with strong performance across all suites. On **RoboCasa**, our model achieves an exceptional **62.1%** success rate, demonstrating robust proficiency in executing everyday household tasks within diverse and unseen kitchen environments. Similarly, on **GR1**, it demonstrates strong proficiency in dexterous humanoid manipulation with a **49.2%** average success rate. When evaluating robustness under environmental perturbations on **LIBERO-plus**, Being-H0.7 secures a **82.1%** zero-shot success rate, which further improves to **84.8%** after fine-tuning on LIBERO-plus, highlighting its resilience against shifted viewpoints, novel textures, and sensor noise. On **RoboTwin 2.0**, Being-H0.7 demonstrates remarkable robustness in complex bimanual manipulation, sustaining an **89.6%** success rate under severe visual domain randomization, with merely a **0.6%** performance drop compared to the clean setting (**90.2%**). Finally, on **CALVIN**, Being-H0.7 proves its capacity for multi-task long-horizon execution and zero-shot environment generalization, successfully completing an average of **4.67** and **4.48** tasks in a row (out of 5) on the ABCD→D and ABC→D splits, respectively.

4.2 Real-world Experiments

We evaluate Being-H0.7 on three real-robot platforms: **PND Adam-U**, **Unitree G1**, and **Franka FR3**. All three platforms are equipped with **Linkerbot O6 (6 DoF)** hands. PND Adam-U and Unitree G1 use bilateral hand configurations, while Franka FR3 provides a single-arm tabletop setting with one external camera and one wrist camera. Figure 4 provides a visual overview of the three deployed embodiments.

4.2.1 Embodiments and Task Suites

Table 3 summarizes the three deployed embodiments. Unless otherwise specified, all embodiments share the same unified control interface and online inference infrastructure. In PND Adam-U, the policy controls 19 body DoF together with bilateral Linkerbot O6 hands. In Unitree G1, the policy exposes a 26-DoF upper-body



(a) PND Adam-U **(b)** Unitree G1 **(c)** Franka FR3

Figure 4: Overview of the real-world embodiments used in this evaluation.

Table 3: **Real-robot embodiments for Being-H0.7.** All evaluated platforms are paired with Linkerbot O6 (6 DoF) hands and share a unified state/action interface.

Platform	Type	Body DoF	Hand	Total DoF	Cameras	Policy Freq.
PND Adam-U	Upper-body humanoid	19	Linkerbot O6 (6 DoF)	31	2 ego-view cameras	20 Hz
Unitree G1	Bimanual humanoid	14	Linkerbot O6 (6 DoF)	26	1 ego-view camera	10 Hz
Franka FR3	Single-arm tabletop	7	Linkerbot O6 (6 DoF)	13	1 external + 1 wrist	20 Hz

action interface, *i.e.*, 14 arm joints plus 12 Linkerbot O6 hand joints. Franka FR3 provides a 7-DoF arm paired with a single Linkerbot O6 hand.

Deployment stack. All three embodiments share the same client-server inference interface. The policy server consumes temporally buffered observations and predicts chunked actions instead of one-step actions. For each query, it returns both robot-space actions for immediate execution and a unified continuation chunk that is used only for the next asynchronous query. This separation is important in practice: the robot executes embodiment-specific commands, while the continuation state remains in a shared representation that keeps the next inference call aligned with the motion that has already been committed.

On the client side, we use a latency-aware **Universal Async Chunking (UAC)** mechanism, implemented as asynchronous real-time chunking. Concretely, the client maintains a thread-safe action buffer together with a running estimate of how many control steps will be consumed before the next chunk becomes available. A control thread pops actions from the committed prefix at the robot frequency, while a parallel inference thread wakes up when the remaining buffer falls below a trigger threshold, fetches the latest observations, and requests the next chunk from the server. The crucial rule is that UAC never rewrites the already committed prefix: it only stitches the future suffix back into the buffer after the estimated inference delay. This *prefix-lock / suffix-update* design absorbs model, transport, and scheduling jitter without changing the policy interface itself, and it makes the same deployment protocol usable across platforms with different control frequencies and embodiment-specific action dimensions.

UAC is the deployment protocol that turns chunked prediction into continuous control. It preserves temporal continuity, reduces visible control stutter, and keeps the evaluation stack uniform across embodiments.

For **Unitree G1**, the policy still exposes the same 26-DoF action interface used by the rest of our deployment stack. The additional backend is a pretrained **AMO** controller [78], used as the balance-aware low-level whole-body module for humanoid execution. In our integration, AMO owns the 50 Hz Unitree body-control loop, predicts lower-body and waist commands conditioned on the latest upper-arm targets, and composes the final body command for execution, while the Linkerbot O6 hands remain controlled through the same hand interface as the other embodiments. This keeps the upper-body policy interface consistent while providing stable whole-body execution on G1.

Table 4: **Real-robot task set for Being-H0.7.** Each task is assigned one primary suite and optional overlap tags; the prompt is the instruction given to the policy during evaluation.

ID	Platform	Primary Suite	Overlap Tags	Prompt
T01	Franka FR3	Dynamic Scene	Motion Reasoning	Catch the fast rolling ball on the table before it leaves the capture area.
T02	PND Adam-U	Physical Reasoning	Long Horizon	Use the pipette to transfer the liquid from the source container into the target container accurately.
T03	Unitree G1	Motion Reasoning	Dynamic Scene, Physical Reasoning	Hit the ball with the racket so that it lands in the marked target area.
T04	PND Adam-U	Physical Reasoning	Long Horizon	Pour the liquid from the beaker through the funnel into the receiving container.
T05	Unitree G1	Dynamic Scene	Motion Reasoning, Physical Reasoning	Pour the objects in the cup into the moving target container.
T06	Franka FR3	Dynamic Scene	Motion Reasoning, Long Horizon, Generalization	Pick the cargo from the moving conveyor and place it on the correct shelf level.
T07	PND Adam-U	Physical Reasoning	Long Horizon, Generalization	Fold the garment neatly into the target folded shape.
T08	Unitree G1	Long Horizon	Dynamic Scene, Motion Reasoning, Generalization	Scan the package on the moving conveyor and sort it to the correct destination.
T09	Unitree G1	Long Horizon	Physical Reasoning, Generalization	Insert the shoe tree into the shoe and place the prepared shoe onto the conveyor.
T10	PND Adam-U	Long Horizon	Dynamic Scene, Generalization	Pick the shoe from the conveyor and pack it into the shoebox.
T11	Franka FR3	Generalization	Long Horizon, Physical Reasoning	Pick the tabletop object and place it into the correct drawer level.
T12	Franka FR3	Physical Reasoning	Motion Reasoning, Long Horizon	Pick up the hammer and drive the nail into the hole.

We design **12 new real-robot tasks** for Being-H0.7 and organize them into five *ability-oriented* suites: **dynamic scene**, **physical reasoning**, **motion reasoning**, **long-horizon execution**, and **generalization**. These suites are intentionally *compositional*: a single task may stress several capabilities at once, such as reacting to moving targets while also reasoning about object trajectories, gravity, containment, or multi-stage subgoals. Table 4 lists the full task set and evaluation prompts. Figure 5 provides a unified visual overview of the 12 task scenes used in this evaluation. For reporting, each task is assigned one primary suite together with optional overlap tags, and suite-level averages are computed over all tasks carrying the corresponding suite tag.

These five suites target distinct difficulty sources. **Dynamic Scene** tasks require the policy to react before a moving object or changing scene leaves the feasible interaction window. **Physical Reasoning** tasks require predicting consequences induced by gravity, fluid transfer, deformable contact, containment, or tool-mediated interaction. **Motion Reasoning** tasks emphasize trajectory anticipation, relative velocity, and contact timing. **Long Horizon** tasks stress subgoal memory and sequential consistency across multiple stages. **Generalization** focuses on preserving task structure under shifted layouts, shelf levels, containers, and object instances.

4.2.2 Evaluation Protocol

We deploy all compared policies through a unified **black-box inference server**. This protocol keeps the surrounding execution stack identical across methods. For each task, we pre-define a set of scene layouts and initial conditions, then randomize both the tested policy endpoint and the rollout order during evaluation. The operator records task success using a fixed binary criterion defined for that task while the active policy endpoint remains hidden. Unless otherwise stated, each task is evaluated over **20 blind trials per method**.

This protocol is especially useful here because several of the new suites explicitly probe *reaction quality* in addition to terminal grasp precision. For example, tasks involving dynamic scene changes, flexible objects, or liquid-like interaction can be sensitive to small timing differences in policy updates. The shared deployment server and fixed blind-evaluation procedure keep those comparisons aligned across methods.

4.2.3 Results Overview

Figure 6 summarizes suite-level success rates computed from the overlap-tag aggregation described above. Because the five suites contain different numbers of tagged tasks, the resulting bars are reported with one decimal place.

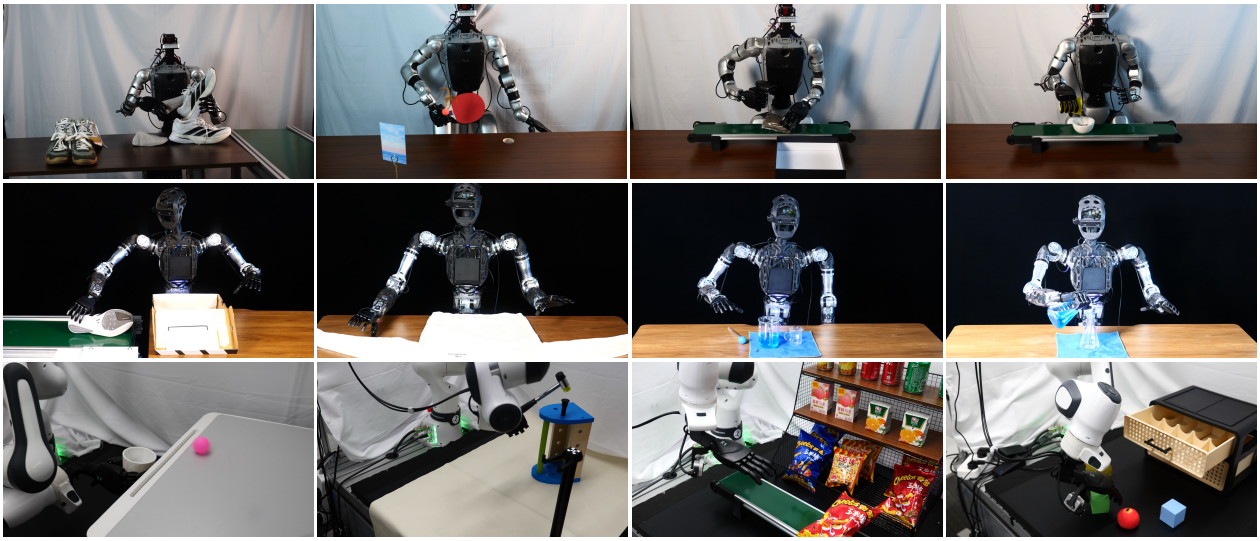


Figure 5: **Visual overview of the 12 real-robot evaluation tasks.** The figure shows the task scenes used in our real-world evaluation across PND Adam-U, Unitree G1, and Franka FR3, covering the five ability-oriented suites.

Figure 6: **Suite-level real-robot success rates (%).** Comparison of Being-H0.7, Being-H0.5, π0.5, and Fast-WAM on the five ability-oriented task suites. Each task is evaluated over 20 blind trials, and each suite score is averaged over all tasks carrying the corresponding suite tag.

Being-H0.7 leads on all five suites, spanning reactive, physical, sequential, and generalization-oriented tasks across all three embodiments. This breadth is the main real-robot result of the section: the improvement appears throughout the benchmark rather than concentrating in one corner of it.

Dynamic and motion-centric tasks are where the predictive advantage is most visible. The clearest margin appears on **Dynamic Scene**, and the same ordering largely carries over to **Motion Reasoning**. These suites contain the most timing-sensitive tasks in the benchmark, including catching a fast rolling ball, racket-based redirection, pouring into a moving receptacle, and conveyor-based interaction. Such tasks punish stale state estimates very quickly: once the object has moved beyond the valid contact window, small pose errors or delayed corrections usually lead to immediate failure. Among the baselines, **Fast-WAM** remains the strongest one in these reactive suites, which is consistent with its emphasis on low-latency action generation. Being-H0.7 extends this advantage further, indicating that reactive success is shaped jointly by runtime responsiveness and by a future-aware latent state that tracks object motion, relative timing, and the downstream consequence of committing to a contact.

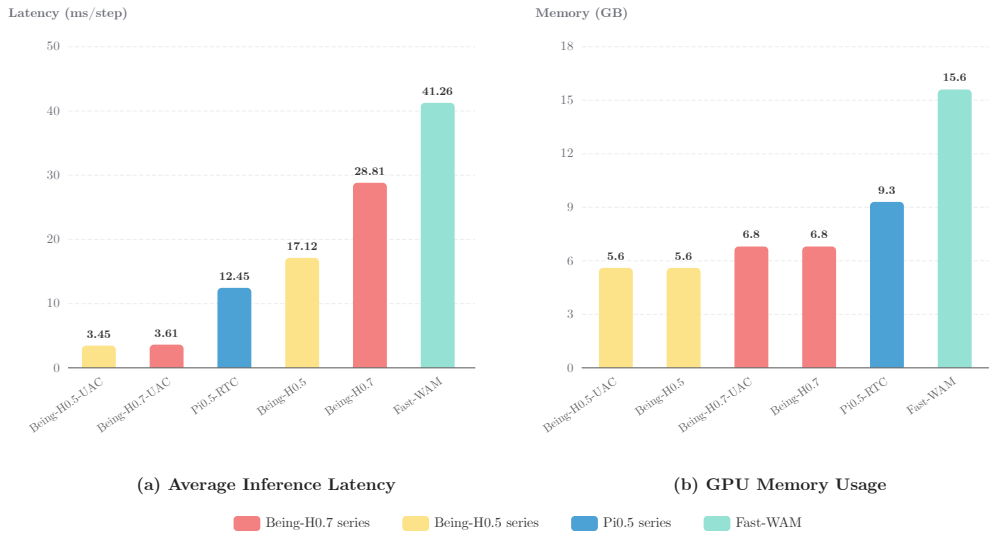


Figure 7: Inference cost measured in the real-world deployment stack. We report it as a system-level view of deployability alongside the main success-rate comparison.

Physical and long-horizon suites highlight a second strength of the model. On **Physical Reasoning** and **Long Horizon**, the closest baseline is **Being-H0.5**, reflecting stronger stable manipulation priors and stage-by-stage execution consistency. Representative tasks here include pipette transfer, funnel pouring, garment folding, shoe-tree insertion, shoe boxing, and hammer-and-nail interaction. Success couples dexterity with reasoning about containment, gravity, deformable contact, tool-mediated force transfer, and how earlier subgoals constrain later ones. Being-H0.7 stays ahead on both suites, showing that the learned world-action prior supports fast reaction and also maintains causal consistency through longer and more physically constrained manipulation chains.

Generalization improvements persist across embodiments. The **Generalization** suite mixes tasks from all three platforms and stresses shifted layouts, shelf heights, containers, object instances, and camera geometries. Here, π**0.5** and **Being-H0.5** remain reasonably competitive, making the suite balanced and practically relevant. Even so, Being-H0.7 remains the most reliable overall, indicating that the benefit of its latent predictive prior carries across changes in embodiment and scene structure. Because the benchmark is intentionally compositional, a consistent lead across all five suite bars is stronger evidence than an isolated win on a single hand-picked task.

Inference infrastructure strengthens real-world deployability. Figure 7 reports system-level inference cost under the same deployment infrastructure. The most visible effect is that the UAC-enabled Being-H variants move into the 3–4 ms/step regime while keeping the same GPU memory footprint as their non-UAC counterparts. This gives the controller more timing slack on dynamic tasks, keeps buffer occupancy steadier under network and scheduler jitter, and makes the online rollout feel substantially smoother at the robot interface. Together with the suite-level results above, the cost plot shows that the policy gains are realized inside an efficient inference loop rather than at the expense of an unwieldy deployment setup.

5 Conclusion

We introduced Being-H0.7, a *latent world-action model* that bridges direct action prediction and world modeling through a compact latent reasoning space. By aligning a deployable prior branch with a future-aware posterior branch, our method injects future-relevant reasoning into action generation without requiring costly pixel-level rollout at inference time. Combined with large-scale human video pretraining, Being-H0.7 provides an effective and scalable framework for embodied models.

References

[1] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[2] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[3] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[4] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[5] J Bjorck Nvidia, Fernando Castaneda, N Cherniadev, X Da, R Ding, L Fan, Y Fang, D Fox, F Hu, S Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

[6] Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Qin Jin, and Zongqing Lu. Being-h0: vision-language-action pretraining from large-scale human videos. *arXiv preprint arXiv:2507.15597*, 2025.

[7] Hao Luo, Ye Wang, Wanpeng Zhang, Sipeng Zheng, Ziheng Xi, Chaoyi Xu, Haiweng Xu, Haoqi Yuan, Chi Zhang, Yiqing Wang, et al. Being-h05: Scaling human-centric robot learning for cross-embodiment generalization. *arXiv preprint arXiv:2601.12993*, 2026.

[8] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[9] Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

[10] Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

[11] Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

[12] Lars Berscheid, Pascal Meißner, and Torsten Kröger. Robot learning of shifting objects for grasping in cluttered environments. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 612–618. IEEE, 2019.

[13] Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

[14] Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. *arXiv preprint arXiv:2307.00595*, 2023.

[15] Nur Muhammad Mahi Shafiullah, Anant Rai, Haritheja Etukuru, Yiqian Liu, Ishan Misra, Soumith Chintala, and Lerrel Pinto. On bringing robots home. *arXiv preprint arXiv:2311.16098*, 2023.

[16] Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[17] 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.

[18] Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, et al. Paligemma 2: A family of versatile vlms for transfer. *arXiv preprint arXiv:2412.03555*, 2024.

[19] Zhiqi Li, Guo Chen, Shilong Liu, Shihao Wang, Vibashan VS, Yishen Ji, Shiyi Lan, Hao Zhang, Yilin Zhao, Subhashree Radhakrishnan, et al. Eagle 2: Building post-training data strategies from scratch for frontier vision-language models. *arXiv preprint arXiv:2501.14818*, 2025.

[20] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[21] Wanpeng Zhang, Zilong Xie, Yicheng Feng, Yijiang Li, Xingrun Xing, Sipeng Zheng, and Zongqing Lu. From pixels to tokens: Byte-pair encoding on quantized visual modalities. In *The Thirteenth International Conference on Learning Representations*, 2025.

[22] Wanpeng Zhang, Yicheng Feng, Hao Luo, Yijiang Li, Zihao Yue, Sipeng Zheng, and Zongqing Lu. Unified multimodal understanding via byte-pair visual encoding. *arXiv preprint arXiv:2506.23639*, 2025.

[23] Luo Hao, Yue Zihao, Zhang Wanpeng, Feng Yicheng, Zheng Sipeng, Ye Deheng, and Lu Zongqing. OpenMMEgo: Enhancing egocentric understanding for LMMs with open weights and data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[24] Yicheng Feng, Yijiang Li, Wanpeng Zhang, Sipeng Zheng, Hao Luo, Zihao Yue, and Zongqing Lu. Videoorion: Tokenizing object dynamics in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20401–20412, 2025.

[25] Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

[26] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[27] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[28] Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024.

[29] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

[30] Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Tian Nian, Liuao Pei, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, et al. Discrete diffusion vla: Bringing discrete diffusion to action decoding in vision-language-action policies. *arXiv preprint arXiv:2508.20072*, 2025.

[31] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

[32] Junjie Wen, Yichen Zhu, Jinming Li, Zhibin Tang, Chaomin Shen, and Feifei Feng. Dexvla: Vision-language model with plug-in diffusion expert for general robot control. *arXiv preprint arXiv:2502.05855*, 2025.

[33] Yifan Zhong, Xuchuan Huang, Ruochong Li, Ceyao Zhang, Zhang Chen, Tianrui Guan, Fanlian Zeng, Ka Num Lui, Yuyao Ye, Yitao Liang, et al. Dexgraspvla: A vision-language-action framework towards general dexterous grasping. *arXiv preprint arXiv:2502.20900*, 2025.

[34] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[35] Michał Zawalski, William Chen, Karl Pertsch, Oier Mees, Chelsea Finn, and Sergey Levine. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

[36] Fanqi Lin, Ruiqian Nai, Yingdong Hu, Jiacheng You, Junming Zhao, and Yang Gao. Onetwovla: A unified vision-language-action model with adaptive reasoning. *arXiv preprint arXiv:2505.11917*, 2025.

[37] Jaden Clark, Suvir Mirchandani, Dorsa Sadigh, and Suneel Belkhale. Action-free reasoning for policy generalization. *arXiv preprint arXiv:2502.03729*, 2025.

[38] Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. *arXiv preprint arXiv:2510.12276*, 2025.

[39] Brent Griffin. Mobile robot manipulation using pure object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 561–571, 2023.

[40] Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pages 5639–5650. PMLR, 2020.

[41] Andreas Ten Pas and Robert Platt. Using geometry to detect grasp poses in 3d point clouds. In *Robotics Research: Volume 1*, pages 307–324. Springer, 2017.

[42] Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025.

[43] Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

[44] Jonas Pai, Liam Achenbach, Victoriano Montesinos, Benedek Forrai, Oier Mees, and Elvis Nava. mimic-video: Video-action models for generalizable robot control beyond vlas. *arXiv preprint arXiv:2512.15692*, 2025.

[45] Yao Feng, Hengkai Tan, Xinyi Mao, Chendong Xiang, Guodong Liu, Shuhe Huang, Hang Su, and Jun Zhu. Vidar: Embodied video diffusion model for generalist manipulation. *arXiv preprint arXiv:2507.12898*, 2025.

[46] Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, et al. Genie envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*, 2025.

[47] Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025.

[48] Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. *arXiv preprint arXiv:2504.02792*, 2025.

[49] Junbang Liang, Pavel Tokmakov, Ruoshi Liu, Sruthi Sudhakar, Paarth Shah, Rares Ambrus, and Carl Vondrick. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*, 2025.

[50] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

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

[52] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

[53] Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. *arXiv preprint arXiv:2512.13030*, 2025.

[54] Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. Videovla: Video generators can be generalizable robot manipulators. *arXiv preprint arXiv:2512.06963*, 2025.

[55] Hao Luo and Zongqing Lu. Learning video-conditioned policy on unlabelled data with joint embedding predictive transformer. In *International Conference on Learning Representations*, 2025.

[56] Aleksandar Vujinovic and Aleksandar Kovacevic. Act-jepa: Novel joint-embedding predictive architecture for efficient policy representation learning. *arXiv preprint arXiv:2501.14622*, 2025.

[57] Ruijie Zheng, Jing Wang, Scott Reed, Johan Bjorck, Yu Fang, Fengyuan Hu, Joel Jang, Kaushil Kundalia, Zongyu Lin, Loïc Magne, Avnish Narayan, You Liang Tan, Guanzhi Wang, Qi Wang, Jiannan Xiang, Yinzhen Xu, Seonghyeon Ye, Jan Kautz, Furong Huang, Yuke Zhu, and Linxi Fan. FLARE: Robot learning with implicit world modeling. In *Annual Conference on Robot Learning*, 2025.

[58] Jingwen Sun, Wenyao Zhang, Zekun Qi, Shaojie Ren, Zezhi Liu, Hanxin Zhu, Guangzhong Sun, Xin Jin, and Zhibo Chen. Vla-jepa: Enhancing vision-language-action model with latent world model. *arXiv preprint arXiv:2602.10098*, 2026.

[59] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 2022.

[60] Weixin Liang, LILI YU, Liang Luo, Srini Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*, 2025.

[61] Randall Balestriero and Yann LeCun. Lejepa: Provable and scalable self-supervised learning without the heuristics. *arXiv preprint arXiv:2511.08544*, 2025.

[62] Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

[63] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

[64] Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. *arXiv preprint arXiv:2510.10274*, 2025.

[65] Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025.

[66] StarVLA Community. Starvla: A lego-like codebase for vision-language-action model developing. *arXiv preprint arXiv:2604.05014*, 2026.

[67] Renming Huang, Chendong Zeng, Wenjing Tang, Jintian Cai, Cewu Lu, and Panpan Cai. Mimic intent, not just trajectories. *arXiv preprint arXiv:2602.08602*, 2026.

[68] Yandan Yang, Shuang Zeng, Tong Lin, Xinyuan Chang, Dekang Qi, Junjin Xiao, Haoyun Liu, Ronghan Chen, Yuzhi Chen, Dongjie Huo, et al. Abot-m0: Vla foundation model for robotic manipulation with action manifold learning. *arXiv preprint arXiv:2602.11236*, 2026.

[69] Wei Wu, Fan Lu, Yunnan Wang, Shuai Yang, Shi Liu, Fangjing Wang, Qian Zhu, He Sun, Yong Wang, Shuailei Ma, et al. A pragmatic vla foundation model. *arXiv preprint arXiv:2601.18692*, 2026.

[70] Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, XinQiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, Li Yi, Wenjun Zeng, and Xin Jin. DreamVLA: A vision-language-action model dreamed with comprehensive world knowledge. In *Annual Conference on Neural Information Processing Systems*, 2025.

[71] Shangchen Miao, Ningya Feng, Jialong Wu, Ye Lin, Xu He, Dong Li, and Mingsheng Long. Jepa-vla: Video predictive embedding is needed for vla models. *arXiv preprint arXiv:2602.11832*, 2026.

[72] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[73] Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

[74] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

[75] Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, et al. Libero-plus: In-depth robustness analysis of vision-language-action models. *arXiv preprint arXiv:2510.13626*, 2025.

[76] Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

[77] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

[78] Jialong Li, Xuxin Cheng, Tianshu Huang, Shiqi Yang, Ri-Zhao Qiu, and Xiaolong Wang. Amo: Adaptive motion optimization for hyper-dexterous humanoid whole-body control. *arXiv preprint arXiv:2505.03738*, 2025.