LaST$_{1.0}$: A Unified Latent World-Action Model with Control-Centric SpatioTemporal Reasoning

***Abstract*— World-Action Models couple action learning with future prediction, enabling robotic policies to capture environmental dynamics for robust control. However, prevailing methods take a reconstruction-oriented approach to future video modeling, allocating substantial capacity to appearance details that are incidental to control. In this work, we reformulate world modeling for manipulation as control-centric future prediction in latent space, and present LaST$_{1.0}$, a unified latent World-Action Model. Rather than modeling an observation-complete video future, LaST$_{1.0}$ predicts compact, geometry-aware latent targets derived from future wrist views, thereby focusing supervision on the evolving robot–object interactions required for fine-grained manipulation. To better support this control-centric paradigm, LaST$_{1.0}$ revisits the visual input interface and attention mechanism of its unified architecture. It adopts a vision-encoder-free design with a lightweight patch embedder, exposing fine-grained visual cues directly to the backbone, and introduces LaST-Attention for structured spatiotemporal interactions across modalities. Evaluations across two robot embodiments, five challenging manipulation tasks, and three simulation benchmarks—RLBench, LIBERO, and LIBERO-Plus—demonstrate that LaST$_{1.0}$ achieves state-of-the-art success rates and generalizes to novel scenarios. These results demonstrate the effectiveness of control-centric latent world modeling for robust manipulation. Anonymous project page: `https://last1-icra.pages.dev`.**

I. INTRODUCTION

Vision-Language-Action (VLA) models have become a strong foundation for generalist robot manipulation by directly mapping visual observations and language instructions to actions [1], [2], [3], [4]. Despite their strong visual and semantic capabilities, the imbalance between dense visual inputs and sparse action supervision can lead to shortcut mappings rather than learning underlying physical dynamics [5], [6]. Consequently, these models can struggle with robust control in complex scenarios that require anticipation of future interactions. World-Action Models (WAMs) address this limitation by coupling action learning with future-state prediction, using future prediction as an auxiliary objective that encourages the shared representation to capture task-relevant changes in the environment [7], [8], [9], [10].

A central question, however, is *what* a policy model should predict about the future. Many existing WAMs instantiate future-state modeling through video generation [5], [7], [11], [9]. While visually expressive, this paradigm allocates substantial capacity to texture, illumination, background appearance, and other photometric details that have little bearing on the next physical interaction [6]. In contrast, fine-grained control depends more directly on localized object motion, relative geometry, contact configuration, and how the interaction region between robot-object evolves over time.

We therefore propose that world modeling for manipulation should be *control-centric* rather than *observation-complete*. For manipulation, the future latent representation should satisfy three properties: (1) preserve the geometric evidence needed for precise action, (2) focus modeling capacity on the region where interaction occurs, and (3) provide predictive supervision during training without becoming mandatory computation at deployment. This perspective changes the role of future prediction: the goal is not to reproduce the world in all observable detail, but to predict the subset of future dynamics that is useful for control. Guided by this principle, we introduce **LaST$_{1.0}$**, a unified latent WAM that jointly optimizes action generation and control-centric future dynamics prediction. Rather than predicting appearance-oriented video latents or visual-semantic image latents, LaST$_{1.0}$ utilizes a geometry-aware encoder [12] to construct compact targets exclusively from future wrist-camera observations, as described in Fig. 1(a). This concentrates the latent supervision on the interaction region, forcing the policy to capture fine-grained relative motions and robot–object geometric relationships. Crucially, our experiments reveal that formulating latent world modeling as a training-time regularizer not only yields improved performance but also enhances closed-loop control efficiency.

To operationalize this control-centric paradigm, we further redesign the visual interface and attention mechanism for LaST$_{1.0}$, as shown in Fig. 1(b) and (c). Semantic vision encoders and reconstruction-oriented VAEs can bias visual representations toward high-level semantics or appearance reconstruction, potentially attenuating the localized visual details vital for manipulation [13], [14]. LaST$_{1.0}$ therefore adopts a vision-encoder-free interface, using a lightweight patch embedder to project raw image patches directly into the unified Transformer backbone, thereby preserving fine-grained visual cues. Within this backbone, rather than relying on conventional attention mechanisms, we introduce LaST-Attention to facilitate the spatiotemporal fusion of multi-modal information across 4D subspaces (token sequence, image height, image width, and relative time), effectively supporting our latent world-action modeling paradigm.

For evaluation, we deploy LaST$_{1.0}$ across two distinct embodiments, utilizing both parallel grippers and dexterous hands across five challenging manipulation tasks. These tasks evaluate the model's ability to perform fine-grained control, spatial reasoning, and long-horizon manipulation. Empirical results demonstrate that LaST$_{1.0}$ achieves state-of-the-art (SOTA) success rates, exhibiting remarkable generalization to novel object, background, and position settings. Moreover,

Manuscript 3964 submitted to 2027 IEEE International Conference on Robotics and Automation (ICRA). Received September 15, 2026.

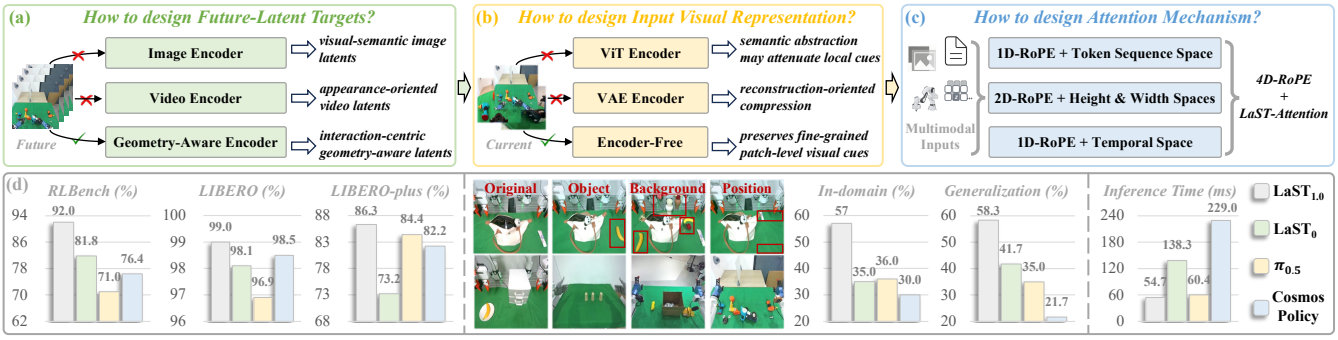

Fig. 1: **Overview of LaST$_{1.0}$.** The top panel illustrates the design roadmap of LaST$_{1.0}$, highlighting the interaction-centric, geometry-aware future-latent supervision, alongside early fusion of visual patches and structured 4D spatiotemporal attention. The bottom panel summarizes the model's performance across simulation, real-world, generalization, and inference latency.

evaluations on three benchmarks—RLBench, LIBERO, and LIBERO-Plus—confirm that our method consistently outperforms prior SOTA baselines, further validating the efficacy of our latent world-action modeling paradigm, as shown in Fig. 1(d). Our contributions are summarized as follows:

- We reformulate world modeling as the prediction of geometry-aware, control-centric future dynamics in latent space, using latent targets derived from future wrist views to capture evolving robot–object interactions.
- We introduce LaST$_{1.0}$, a latent world-action model that natively unifies vision-language understanding, latent world modeling, and action generation. Within this model, we design LaST-Attention to facilitate structured spatiotemporal fusion of multimodal information.
- Experiments demonstrate that LaST$_{1.0}$ achieves SOTA performance across two embodiments, five challenging manipulation tasks, and three simulation benchmarks, together with robust generalization to novel scenarios.

II. RELATED WORK

A. Vision-Language-Action and World-Action Models

Generalist robot policies have primarily evolved along two complementary paradigms. Vision-language-action (VLA) models directly map visual observations and language instructions to robot actions [1], [2], [3], [15]. World-action models (WAMs) additionally model future visual dynamics, either jointly with action prediction or through an imagine-then-act formulation, to provide predictive priors for control [8], [16], [17], [11]. To focus on more informative representations and improve inference efficiency, recent works have explored world modeling in latent spaces [6], [18], [19]. However, these approaches often emphasize scene-level representations rather than information directly relevant to robot-object interaction and manipulation control. LaST$_{1.0}$ instead focuses latent world modeling on control-centric information by constructing geometry-aware latent targets that capture the evolving robot-object interaction, providing more relevant predictive supervision for action generation.

B. Geometric Representation for Robotic Manipulation

Geometric information is important for manipulation, where precise actions depend on spatial relations and robot-object configurations. Recent works incorporate 3D geometry into manipulation policies either explicitly through depth [20], [21], point clouds [22], [23], [24], [25], or implicitly through geometric representations and auxiliary supervision [26], [27], [28]. These methods primarily use geometry from the current observation to enhance the policy's spatial perception. More recently, WAMs have begun to model geometric dynamics beyond video prediction, incorporating future depth to improve spatial understanding and manipulation [29]. In contrast, LaST$_{1.0}$ models future geometry in a compact latent space, focusing on the interaction region and control-centric geometry-aware dynamics rather than reconstructing scene-level video or depth observations.

III. METHOD

To better understand the overall framework of LaST$_{1.0}$, we first introduce the model architecture and LaST-Attention in Sec. III-B and Sec. III-C, followed by the implementation of control-centric latent world modeling in Sec. III-D.

A. Problem Formulation

We formulate vision-language robotic manipulation as learning a policy π_θ via imitation learning on expert demonstrations D. At each timestep t, given the language instruction l and multi-view observations $I_t = \{I_t^k\}_{k=1}^K$, the policy predicts an action chunk $\mathbf{a}_{t+1:t+H}$ over a horizon H. Each action encodes an embodiment-specific control command, comprising a relative end-effector pose coupled with either a parallel gripper state or a dexterous multi-finger configuration. To regularize action learning, we formulate future observations as an auxiliary latent world-modeling target (detailed in Sec. III-D). This predictive supervision acts as a training-time regularizer, embedding physical priors into the representation while allowing the policy π_θ to execute closed-loop control based solely on (l, I_t) at inference.

B. LaST$_{1.0}$ Architecture

LaST$_{1.0}$ features an encoder-free unified architecture, projecting multimodal signals directly into the backbone to preserve fine-grained visual details critical for manipulation. To facilitate effective multimodal fusion without overwhelming

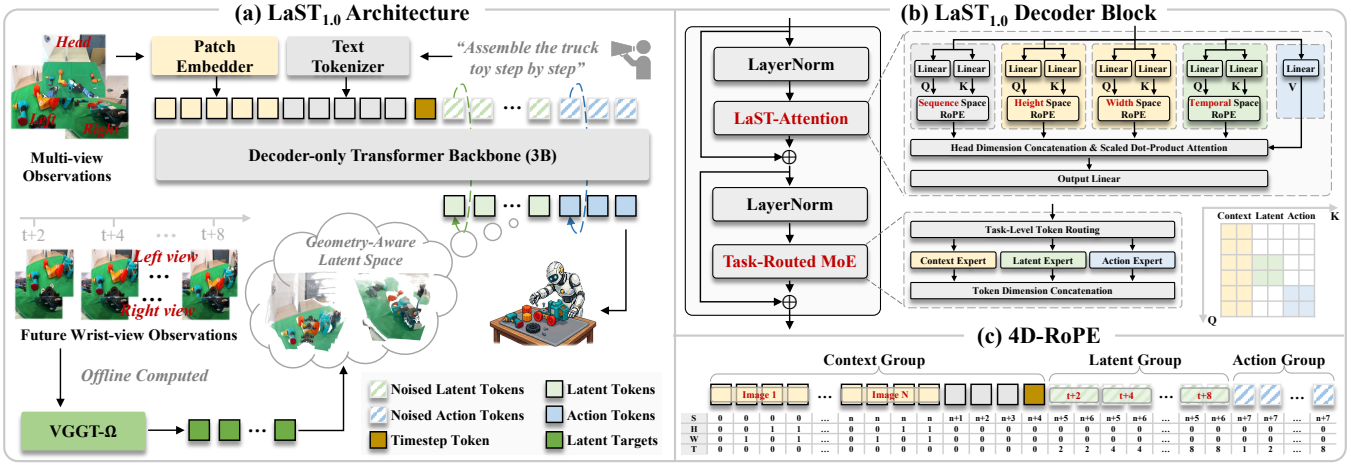

Fig. 2: **LaST$_{1.0}$ Framework. (a)** LaST$_{1.0}$ employs an encoder-free unified backbone that takes projected multimodal context to jointly predict flow velocities for latent and action tokens. The latent targets are extracted via VGGT-Ω from future wrist views to enable interaction-centric geometry-aware reasoning. **(b)** Each decoder block in LaST$_{1.0}$ features LaST-Attention, which independently rotates queries and keys across token sequence, image height, image width, and time axes to enhance spatiotemporal fusion. A subsequent Task-Routed MoE effectively expands model capacity. **(c)** Multimodal Tokens are represented in a 4D space with separate sequence, spatial and temporal coordinates, preserving their inherent structure.

the backbone's capacity, we employ task-routed experts ensuring stable and decoupled learning across vision-language understanding, latent world modeling, and action execution.

1) Encoder-Free Multimodal Projection: The language instruction l is first encoded by the pretrained Qwen3-1.7B tokenizer [30] and mapped into embeddings $f^{\text{lang}} \in \mathbb{R}^{N_{\text{lang}} \times d_h}$ with a hidden dimension of $d_h = 2048$. For visual inputs, while conventional deep vision encoders typically yield tokens that focus on high-level semantic information [13], [31], precise robotic manipulation relies heavily on fine-grained local details. To better capture this information, LaST$_{1.0}$ bypasses deep vision encoders in favor of a lightweight projection interface, as shown in Fig. 2 (a). Specifically, raw images $I \in \mathbb{R}^{H \times W \times 3}$ are processed by a two-layer convolutional module, which focuses on extracting local spatial details and directly projects them into the backbone's latent space, yielding visual tokens $f^{\text{img}} \in \mathbb{R}^{N_{\text{img}} \times d_h}$. These visual tokens are concatenated with the language tokens and a flow-matching timestep token to form the **context** group. Furthermore, the noised future-latent targets and action chunks are independently projected to form the **latent** and **action** groups. Logically, a single training sample is composed of these three distinct token groups: $[\mathbf{X}^{\text{context}}; \mathbf{X}^{\text{latent}}; \mathbf{X}^{\text{action}}]$.

2) Shared Backbone with Task-Routed MoE: The backbone of LaST$_{1.0}$ is a 20-layer Transformer architecture, initialized with the first 20 layers of Qwen3-1.7B [30]. To efficiently scale model capacity, we replicate the standard Feed-Forward Network (FFN) in each layer threefold to construct a task-routed Mixture-of-Experts (MoE), as shown in Fig. 2 (b). Specifically, the pretrained FFN is cloned into *context*, *latent*, and *action* experts, deterministically routed to process their corresponding tokens. This design expands model capacity while maintaining the cost of a single expert.

3) Latent and Action Decoders: Separate final layer normalizations and two-layer MLP heads decode the latent and action states into flow velocities: $\widehat{\mathbf{V}}_A = G_A(\mathbf{H}^A)$ and $\widehat{\mathbf{V}}_Z = G_Z(\mathbf{H}^Z)$. Here, G_A and G_Z are branch-specific heads for action and future-latent tokens, respectively.

C. LaST-Attention with 4D-RoPE

While our task-routed MoE effectively decouples FFN representations, applying a similar routing to attention layers segregates tokens and hinders cross-modal interaction. To ensure comprehensive multimodal fusion, we propose LaST-Attention with 4D-Rotary Positional Embedding (RoPE). This design endows each token with four distinct positional dimensions and aggregates attention computations across these independent subspaces, facilitating a deep, structurally-aware information exchange across modalities.

1) 4D-RoPE: As shown in Fig. 2 (c), each token is assigned a 4D coordinate tuple $\mathbf{p} = (n, y, x, \tau)$ corresponding to its sequence order, spatial height, spatial width, and relative time. This coordinate allocation is strictly governed by token semantics: Language tokens utilize only the sequence axis $(n, 0, 0, 0)$. Visual patches from the same image share a base sequence index and temporal offset τ, but logically span the 2D grid (y, x). Crucially, the generative latent and action tokens are assigned distinct combinations of sequence and temporal coordinates to uniquely identify their future slots and action horizons. This structured 4D allocation provides a stable coordinate convention across modalities and structurally prevents variable-length sequential inputs from distorting the underlying spatiotemporal relationships.

2) LaST-Attention: To effectively process the 4D coordinates assigned to each token, LaST-Attention partitions every attention head into four independent subspaces. Specifically, the queries and keys within a single head are projected into subspaces corresponding to the axis set $\phi = \{\text{S}, \text{H}, \text{W}, \text{T}\}$, which explicitly map to the sequence (n), height (y), width (x), and temporal (τ) dimensions of the coordinate tuple

p. Axis-specific RoPE is then applied independently to each subspace based on its corresponding coordinate value. Recognizing that different axes operate on vastly different scales, we assign distinct base frequencies (β) to each RoPE axis. The sequence axis inherits the large base ($\beta_{\mathrm{S}} = 10^6$) from the LLM, while the spatial and temporal axes utilize progressively smaller bases ($\beta_{\mathrm{H,W}} = 10^4, \beta_{\mathrm{T}} = 10^2$) to reflect the substantially shorter coordinate ranges of image grids and action horizons. After independent rotation, the four subspaces are concatenated along the head dimension to reconstruct the full query and key representations. The final attention logit between token i and j is thus formulated as:

$$e_{ij} = \frac{1}{\sqrt{d_{\text{head}}}} \sum_{\alpha \in \phi} \left\langle \overline{\mathbf{q}}_i^{\alpha}, \overline{\mathbf{k}}_j^{\alpha} \right\rangle + M_{ij}, \tag{1}$$

where $\overline{\mathbf{q}}^{\alpha}$ and $\overline{\mathbf{k}}^{\alpha}$ denote the 4D-RoPE-injected subspace representations, and $\mathbf{M}$ is the attention mask (as shown in Fig. 2 (b)). Specifically, $M_{ij} = 0$ if token i is allowed to attend to token j, and $-\infty$ otherwise. These logits are subsequently passed through a softmax function to weight the value representations $\mathbf{v}$. Consequently, the final aggregated output of each attention head jointly reasons over relative sequence order, 2D spatial displacement, and temporal offset.

D. Control-Centric Latent World Modeling

While existing WAMs [7], [11] typically predict future video latents, our *control-centric* formulation instead models a compact future interaction state. This approach retains information relevant to manipulation while discarding incidental appearance variation. Our latent targets exhibit three core properties: **(1) geometry-aware**, preserving the spatial structure that governs object motion and contact; **(2) interaction-centric**, focusing on the localized region where manipulation occurs; and **(3) inference-independent**, providing privileged future supervision during training without becoming a computational dependency at deployment.

1) Geometry-Aware Future Targets: Rather than conditioning the policy on explicit 3D point clouds, we construct a compact implicit geometric target using a frozen VGGT-Ω [12]. Its register tokens provide a compact representation of the spatial structure within visual observations. By jointly encoding multiple future frames, the model produces temporally fused, geometry-aware latents instead of independent per-frame features. Crucially, rather than treating VGGT-Ω as a spatial feature extractor for *current* observations, we repurpose it to construct geometry-aware *future* targets.

2) Interaction-Centric Wrist World: To reduce supervision from global background regions, we restrict future modeling to the *wrist-world*, i.e., short-horizon future observations captured by the wrist cameras. This perspective centers the predictive target on the evolving geometric interactions between the end-effectors and manipulated objects. By default, the action-chunk horizon is $H = 8$, from which we uniformly sample $F = 4$ future timesteps at offsets $\delta_f = \frac{fH}{F} = 2f$ for $f = 1, \ldots, F$, corresponding to $\{t+2, t+4, t+6, t+8\}$. For each arm, we collect its wrist-view frames across these F sampled timesteps and input them together into the frozen VGGT-Ω. The encoder extracts $R = 16$ register tokens per frame, yielding a temporal latent representation of $16 \times F$ tokens for this specific arm, denoted as $\mathbf{Z}^{\text{arm}}$. Crucially, for dual-arm setups, we independently process this multi-frame wrist sequence for each arm, and then concatenate their respective latents along the token dimension to construct the complete future-latent target $\mathbf{Z} = [\mathbf{Z}^{\text{left}}, \mathbf{Z}^{\text{right}}]$.

3) Training-Only Latent Regularization: Inspired by [11], we further design a branch-decoupled attention mask $\mathbf{M}$, as shown in Fig. 2(b), to formulate this latent world modeling as a training-time regularizer. Since these geometry-aware targets are derived from the frozen VGGT-Ω, they can be precomputed offline, ensuring this auxiliary objective introduces negligible additional computational overhead during training. Consequently, LaST$_{1.0}$ can directly output actions based on the input context during inference, bypassing the latent prediction and guaranteeing deployment efficiency.

E. Training and Inference

We normalize all action dimensions to $[-1, 1]$ using dataset quantile bounds, while applying z-normalization to the target latents. Both the action and latent branches are trained jointly via linear conditional flow matching [32]. Let $u \sim \text{Beta}(1.5, 1.0)$ denote the flow timestep, and $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ be standard Gaussian noise. For a given generative target $\mathbf{Y} \in \{\mathbf{A}, \mathbf{Z}\}$, we construct the noised input $\mathbf{Y}_u$ and the target velocity $\mathbf{V}_Y$ as: $\mathbf{Y}_u = (1-u)\mathbf{Y} + u\boldsymbol{\varepsilon}, \mathbf{V}_Y = \boldsymbol{\varepsilon} - \mathbf{Y}$, where $\mathbf{A}$ represents the action chunk $\mathbf{a}_{t+1:t+H}$ and $\mathbf{Z}$ denotes the latent targets. The backbone then predicts the corresponding flow velocities $\widehat{\mathbf{V}}_A$ and $\widehat{\mathbf{V}}_Z$. We optimize the entire network by minimizing the Mean Squared Error (MSE) between the predicted and target velocities:

$$\mathscr{L} = \mathbb{E}_{u,\varepsilon} \left[\left\| \widehat{\mathbf{V}}_A - \mathbf{V}_A \right\|_2^2 + \lambda_{\text{lat}} \left\| \widehat{\mathbf{V}}_Z - \mathbf{V}_Z \right\|_2^2 \right], \tag{2}$$

where $\lambda_{\text{lat}} = 0.1$ weights the latent regularization.

At inference, following the decoupled topology, the latent pathway is omitted. Starting from pure Gaussian noise $\mathbf{A}_{u=1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we synthesize the action chunk by querying the action vector field $\widehat{\mathbf{V}}_A$ and solving the flow ODE toward $u = 0$ using standard Euler integration with $K = 4$ steps.

IV. EXPERIMENT

A. Simulation Experiment

1) Benchmarks: We evaluate LaST$_{1.0}$ on the RLBench and LIBERO benchmarks to comprehensively validate its manipulation capabilities. For **RLBench** [35], we select 10 diverse tasks in CoppeliaSim utilizing a Franka Panda arm. Following standard protocols [2], [15], we extract keyframes from 100 expert trajectories per task. For **LIBERO** [36], we utilize the official training datasets provided for its four suites (*Spatial*, *Object*, *Goal*, and *Long*). Each suite comprises 10 distinct tasks with 50 official expert demonstrations per task, totaling 500 trajectories per suite. Additionally, to assess generalization, we also evaluate on the **LIBERO-Plus** [37]. For this evaluation, we directly apply the weights trained on the LIBERO datasets without any further fine-tuning.

TABLE I: **Results on RLBench.** All methods are trained in the multi-task setting, and we report average success rates (%).

Models	Close box	Close laptop lid	Toilet seat down	Sweep to dustpan	Close fridge	Phone on base	Take umbrella	Take frame off hanger	Place wine at rack	Water plants	Average
OpenVLA-OFT [3]	96.0	70.0	80.0	50.0	84.0	38.0	44.0	20.0	38.0	26.0	54.6
$\pi_{0.5}$ [1]	90.0	96.0	94.0	84.0	**100.0**	18.0	28.0	**86.0**	76.0	38.0	71.0
HybridVLA [2]	92.0	96.0	**100.0**	90.0	96.0	44.0	58.0	68.0	48.0	44.0	73.6
Fast-WAM [11]	94.0	90.0	96.0	74.0	**100.0**	30.0	66.0	56.0	8.0	14.0	62.8
Cosmos Policy [9]	94.0	94.0	**100.0**	60.0	76.0	78.0	**80.0**	70.0	54.0	58.0	76.4
LaST_0 [4]	96.0	92.0	**100.0**	82.0	86.0	74.0	78.0	66.0	80.0	64.0	81.8
$\text{LaST}_{1.0}$ (Ours)	**100.0**	**100.0**	**100.0**	**96.0**	**100.0**	**86.0**	78.0	66.0	**98.0**	**96.0**	**92.0**

TABLE II: **Results on LIBERO and LIBERO-Plus.** We perform three test runs and report the average success rate (%).

Models	LIBERO					LIBERO-Plus							
	Spatial	Object	Goal	Long	Average	Camera	Robot	Language	Light	Background	Noise	Layout	Average
OpenVLA-OFT [3]	97.6	98.4	97.9	94.5	97.1	56.4	31.9	79.5	88.7	**97.3**	75.8	74.2	70.0
$\pi_{0.5}$ [1]	98.8	98.2	98.0	92.4	96.9	78.4	73.6	80.8	96.2	94.1	89.0	84.5	84.4
StarVLA-α [33]	99.0	99.8	98.5	94.1	97.9	48.7	63.4	86.8	95.8	94.6	75.0	80.2	76.1
Fast-WAM [11]	98.2	**100.0**	97.0	95.2	97.6	16.4	44.5	68.9	78.2	53.7	37.7	60.7	50.0
Cosmos Policy [9]	98.1	**100.0**	98.2	97.6	98.5	75.8	63.3	81.7	**96.5**	88.9	92.7	82.2	82.2
LingBot-VA [34]	98.5	99.6	97.2	**98.5**	98.5	40.9	**83.0**	86.4	82.3	53.1	64.4	76.2	69.5
VLA-JEPA [18]	96.2	99.6	97.2	95.8	97.2	63.3	67.1	85.4	95.6	93.6	66.3	85.1	78.0
LaST_0 [4]	**99.2**	99.6	98.0	95.6	98.1	**81.4**	36.5	80.2	86.4	85.7	79.8	68.9	73.2
$\text{LaST}_{1.0}$ (Ours)	98.5	99.8	**99.1**	**98.5**	**99.0**	77.9	68.9	**89.4**	94.7	93.1	**97.3**	**87.3**	**86.3**

2) Training and Evaluation Protocol: We benchmark $\text{LaST}_{1.0}$ against recent state-of-the-art Vision-Language-Action (VLA) models, including OpenVLA-OFT [3], $\pi_{0.5}$ [1], HybridVLA [2], StarVLA-α [33], and LaST_0 [4], as well as representative World-Action Models (WAMs), including Fast-WAM [11], Cosmos Policy [9], and LingBot-VA [34]. $\text{LaST}_{1.0}$ is trained using AdamW with a learning rate of 1×10^{-4} on 8 NVIDIA RTX PRO 5000 GPUs. We train the model for 40k steps on RLBench and 80k steps on LIBERO. All input images are resized to 256×256. RLBench uses a single front-view, whereas LIBERO uses both agent-view and wrist-view observations. During evaluation, we test the final checkpoint of each method. On RLBench, we perform 50 independent rollouts per task and report the success rate. On both LIBERO and LIBERO-Plus, we evaluate $\text{LaST}_{1.0}$ across three independent test runs, utilizing 50 rollouts per task for LIBERO and following the standard category-specific evaluation protocol for LIBERO-Plus.

3) Results: As shown in Table I and Table II, $\text{LaST}_{1.0}$ establishes a new state-of-the-art across all evaluated benchmarks. On RLBench, it achieves the highest average success rate of 92.0%, outperforming the strongest baseline, LaST_0 (81.8%), by 10.2 percentage points and eclipsing WAMs like Cosmos Policy (76.4%). By scoring near-perfect success rates ($\geq 96.0\%$) on seven out of ten tasks, it shows that our encoder-free, geometry-aware reasoning preserves fine-grained spatial precision—an area where semantic-heavy VLAs (e.g., OpenVLA-OFT at 54.6%) inherently struggle. On the LIBERO benchmark, $\text{LaST}_{1.0}$ yields a leading 99.0% average, notably maintaining 98.5% on the challenging *Long* suite, which demonstrates that our branch-decoupled regularization effectively injects geometric priors within the policy representation. Finally, under zero-shot transfer to LIBERO-Plus, $\text{LaST}_{1.0}$ delivers peak out-of-distribution robustness with the highest average of 86.3%. It excels across severe shifts like *Sensor Noise* (97.3%) and *Language* (89.4%), whereas baselines reliant on space-discretizing VAEs (Fast-WAM at 50.0%) suffer severe degradation. Crucially, $\text{LaST}_{1.0}$ significantly surpasses LaST_0 (73.2%) by a striking margin of 13.1 percentage points on LIBERO-Plus. This substantial improvement highlights that our proposed interaction-centric, geometry-aware latent world modeling provides far more robust and actionable cues for control than previous designs. These results demonstrate that learning interaction-aware geometric dynamics yields superior generalization over observation-complete video prediction.

B. Ablation Study

We conduct controlled ablations on the same ten-task RLBench protocol, changing one factor at a time while keeping all other settings fixed.

1) Latent World Modeling Design: **Choice of latent targets.** We compare different future latent targets under a controlled policy interface, with target encoders frozen. VGGT-Ω uses its native 16 registers, while alternative representations are pooled to 16 tokens per future timestep and projected to the same backbone dimension. All other policy and optimization settings remain fixed. As shown in Fig. 3(a), VGGT-Ω [12] register targets achieve 92.0% success, outperforming pooled full VGGT-Ω features (89.6%), Uni3D-g [38] point-cloud latents (87.8%), Wan 2.2 VAE [39] video latents (85.4%), DINOv3-H+ [40] image latents (83.6%), and no latent supervision (81.8%). Compared with the scene-level future point-cloud target used in LaST_0 [4], our method improve success by 4.2 points without explicit point-cloud inputs. **Geometric content of predicted latents.** To assess whether the predicted future latents encode control-relevant geometry, we perform timestep-aligned linear probing. For each variant, we freeze the policy and fit a linear ridge probe on held-out demonstrations to predict the relative position between the end-effector and a contact-relevant object point at each predicted future timestep, $r_{t+\tau} = p^{\text{obj}}_{t+\tau} - p^{\text{ee}}_{t+\tau}, \tau \in$

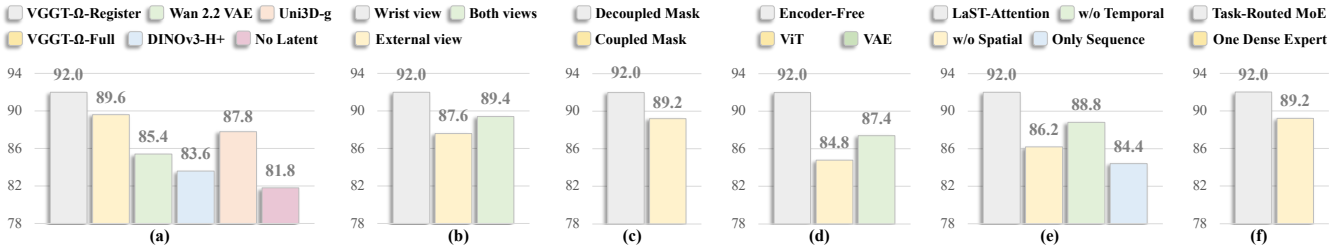

Fig. 3: **Ablation studies of LaST$_{1.0}$ on RLBench. (a–c)** Ablations on the latent world modeling design, including future latent target, observation views, and the branch-decoupled mask. **(d–f)** Ablations on the architecture design, comparing the encoder-free interface, LaST-Attention with 4D-RoPE, and the Task-Routed MoE. We report the average success rate (%).

TABLE III: **Future latent probing results on RLBench.**

	VGGT-Ω-Register	VGGT-Ω-Full	Uni3D-g	Wan2.2 VAE	DINOv3
Error (cm) ↓	**0.23**	0.29	0.77	1.88	2.69

TABLE IV: **Hyperparameter Ablations on RLBench.** We ablate the latent loss weight λ_{lat} and the sampled frames F.

	Latent Loss Weight λ_{lat}				Future Frames F			
Value	0	0.1	0.5	1.0	1	2	4	8
Success rates (%)	81.8	**92.0**	89.2	87.8	88.2	89.8	**92.0**	90.4

$\{1,\ldots,F\}$, from the corresponding predicted latent. Here, F denotes the number of predicted future timesteps, and each latent is probed against the geometry at its matched timestep under a common coordinate frame. We report the three-dimensional Euclidean error over the predicted future timesteps. This metric directly captures the robot–object interaction geometry relevant to manipulation. As shown in Table III, VGGT-Ω registers achieve the lowest error (0.23 cm), followed by pooled VGGT-Ω full features (0.29 cm), Uni3D-g (0.77 cm), Wan 2.2 VAE (1.88 cm), and DINOv3-H+ (2.69 cm). The error ordering agrees with manipulation success, suggesting that future latent target choice affects the accessibility of manipulation-relevant spatial relations.

Observation views of latent targets. Figure 3(b) shows that wrist-only targets (92.0%) outperform external-view (87.6%) and both-view (89.4%) targets. Together with the probing results, this supports concentrating future supervision on the robot–object interaction region. Adding an external view does not improve performance in this setting, suggesting that broader scene coverage is not necessarily a more effective auxiliary prediction target for manipulation. **Predictive supervision and deployment.** We also distinguish future prediction from current-state representation supervision. Under the single-timestep setting, predicting a future geometry-aware latent achieves 88.2% success ($F = 1$ in Table IV), compared with 82.4% when the target is extracted from the current observation. This result supports the value of predictive supervision beyond aligning the policy with current geometric features. Finally, the decoupled mask achieves 92.0% success compared with 89.2% for coupled attention (Fig. 3(c)). The decoupled design allows future prediction to regularize policy learning while keeping action generation independent of future-latent tokens at deployment. Thus, the policy benefits from predictive training supervision while retaining action-only inference.

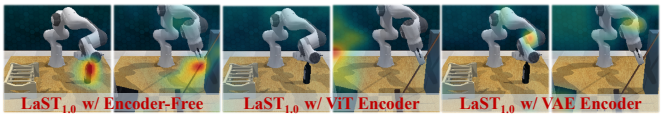

Fig. 4: **Action-to-Vision Attention Maps.** Unlike ViT and VAE approaches, our encoder-free LaST$_{1.0}$ avoids attention diffusion and keeps sharp focus on task-critical regions.

2) Architecture Design: Regarding the visual input representation, our encoder-free interface achieves 92.0%, outperforming a SigLIP2 [41] vision encoder (84.8%) and a reconstruction-oriented Wan2.2 VAE [39] (87.4%) as shown in Fig. 3(d). Correspondingly, attention visualizations in Fig. 4 reveal that our encoder-free design yields localized action-to-vision attention around manipulated objects, confirming its ability to preserve fine-grained control cues. We next evaluate LaST-Attention with 4D-RoPE in Fig. 3(e). The full 4D formulation achieves 92.0%, while removing the spatial (2D) or temporal (3D) space reduces performance to 86.2% and 88.8%, respectively. Using only the token sequence space (1D) further drops performance to 84.4%. Finally, the Task-Routed MoE (92.0%) outperforms a dense expert with matched parameters (89.2%; Fig. 3(f)), confirming that the performance gain stems from decoupling role-specific computations rather than expanding model capacity.

3) Hyperparameters Design: Table IV shows that a moderate latent weight performs best. $\lambda_{\text{lat}} = 0.1$ achieves 92.0%, compared with 81.8%, 89.2%, and 87.8% for 0, 0.5, and 1.0. With this weight fixed, using $F = 1, 2, 4$, and 8 future frames gives 88.2%, 89.8%, 92.0%, and 90.4%, respectively. We therefore adopt $\lambda_{\text{lat}} = 0.1$ and $F = 4$ for LaST$_{1.0}$ by default.

C. Real-World Experiment

1) Data Collection and Task Definition: To evaluate LaST$_{1.0}$ across diverse embodiments and task horizons, we design five real-world multi-stage manipulation tasks. Data collection strategies are tailored to the control complexity of each task. **(1) Dual-arm Franka with Grippers:** We utilize a SpaceMouse to collect 100 expert trajectories per task. The evaluation includes two tasks: *Put Object in Bag* (S1: place the object inside, and S2: zip the bag) and *Put Object in Drawer* (S1: open the top drawer, S2: place the object inside, and S3: close the drawer). **(2) Dual-arm Tianji with Grippers:** For fine-grained, long-horizon manipulation, we collect 300 demonstrations per task via a PICO VR teleoperation platform. The tasks are *Parcel-Box Packing*

TABLE V: **Results on Real-World Tasks.** We decompose each task into subtasks and record the success rates (%). Speed (ms) denotes the latency of a single inference, evaluated on an NVIDIA RTX 4090 GPU under identical settings.

Method	Franka + Gripper					Tianji + Gripper							Tianji + Hand		Stage Average	Task Average	Speed (ms)
	Put Object in Bag		Put Object in Drawer			Parcel-Box Packing			Assemble Truck Toy				Stack Cups				
	S1	S2	S1	S2	S3	S1	S2	S3	S1	S2	S3	S4	S1	S2			
$\pi_{0.5}$ [1]	75.0	50.0	**90.0**	70.0	65.0	90.0	70.0	20.0	**85.0**	10.0	5.0	5.0	60.0	40.0	52.5	36.0	60.4
Cosmos Policy [9]	80.0	55.0	85.0	45.0	35.0	70.0	60.0	10.0	70.0	5.0	0.0	0.0	60.0	**50.0**	44.6	30.0	229.0
LaST$_0$ [4]	80.0	60.0	**90.0**	60.0	60.0	75.0	70.0	15.0	75.0	20.0	15.0	5.0	65.0	35.0	51.8	35.0	138.3
LaST$_{1.0}$ (Ours)	**95.0**	**70.0**	**90.0**	**80.0**	**75.0**	**95.0**	**90.0**	**55.0**	**85.0**	**40.0**	**35.0**	**35.0**	**75.0**	**50.0**	**69.3**	**57.0**	**54.7**

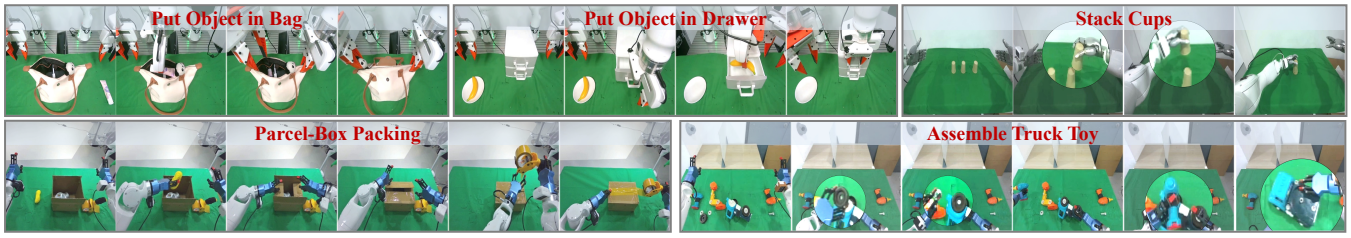

Fig. 5: **Visualization of LaST$_{1.0}$ performing diverse tasks across different robotic embodiments in real-world executions.**

TABLE VI: **Generalization Analysis.** Task-level success rates (%) evaluated on the two stages of "Put Object in Bag".

Model	Original	Object	Background	Position
$\pi_{0.5}$ [1]	50.0	35.0 (↓ 30%)	50.0 (↓ 0%)	20.0 (↓ 60%)
Cosmos Policy [9]	55.0	20.0 (↓ 64%)	35.0 (↓ 36%)	10.0 (↓ 82%)
LaST$_0$ [4]	60.0	45.0 (↓ 25%)	45.0 (↓ 25%)	35.0 (↓ 42%)
LaST$_{1.0}$ (Ours)	**70.0**	**55.0** (↓ 21%)	**70.0** (↓ 0%)	**50.0** (↓ 29%)

(S1: place the item into the box, S2: fold the four flaps, and S3: seal the box with tape) and *Assemble Truck Toy* (S1: position the wheel, S2: place the white accessory, S3: fasten with an electric drill, and S4: flip the truck upright). **(3) Dual-arm Tianji with Sharpa Hands:** To validate dexterous hand control, we capture 100 trajectories using Manus motion-capture gloves for the *Stack Cups* task (S1: stack the right cup onto the middle cup, followed by the left cup (S2)).

2) Training and Evaluation Protocol: We compare LaST$_{1.0}$ against three state-of-the-art baselines: two VLA models ($\pi_{0.5}$ [1] and LaST$_0$ [4]) and one WAM (Cosmos Policy [9]). For a rigorous and fair comparison, all policies receive identical visual inputs consisting of three camera streams: two wrist views and one static third-person exterior view, all resized to 256×256 resolution. All baseline policies are trained on our exact real-world demonstration datasets using their officially released training scripts. LaST$_{1.0}$ is trained for 10k steps per task in real-world experiments, using the same setting as in the simulation. During evaluation, we test the multi-stage robustness of each policy. Every task is evaluated across 20 real-world rollouts under varying initial object poses and spatial arrangements. Crucially, to provide a granular analysis of long-horizon capabilities, we decompose every task into its constituent stages (as defined in Sec. IV-C.1) and explicitly record the success rate for each subtask. We report both the stage-level and task-level average success rates, which are calculated by averaging across all individual stages and across all evaluated tasks, respectively.

3) Quantitative and Qualitative Results: As shown in Table V and visualized in Fig. 5, LaST$_{1.0}$ establishes a new state-of-the-art across all real-world tasks. It achieves stage-level and task-level average success rates of 69.3% and 57.0%, respectively, outperforming $\pi_{0.5}$ (52.5%/36.0%), LaST$_0$ (51.8%/35.0%), and Cosmos Policy (44.6%/30.0%). This substantial gap in task-level metrics directly underscores LaST$_{1.0}$'s robustness against compounding errors in long-horizon tasks. In the demanding four-stage *Assemble Truck Toy* task, baselines suffer an abrupt performance collapse at the second stage (S2 success rates plummet to $\leq 20.0\%$) due to strict fine-grained manipulation requirements, whereas LaST$_{1.0}$ robustly secures 40.0%. This highlights how our interaction-centric, geometry-aware targets preserve precision. Furthermore, LaST$_{1.0}$ also adapts well to high-DoF continuous control, matching or exceeding baseline performances on the *Stack Cups* task utilizing Sharpa hands. Beyond execution success, LaST$_{1.0}$ operates at an efficient 54.7 ms per step. This latency is lower than LaST$_0$ (138.3 ms) and Cosmos Policy (229.0 ms), highlighting the efficiency of our encoder-free design and inference-independent latent world modeling, which reduce visual processing overhead while preserving fine-grained cues for real-time control. More visualizations are available on our project page.

4) Generalization Analysis: To evaluate real-world robustness, we design three out-of-distribution (OOD) scenarios for the dual-arm "Put Object in Bag" task (Fig. 1(d)). Specifically, we introduce **(1) Object Generalization** by replacing the training object (facial cleanser) with a novel item of different shape and texture (banana); **(2) Background Generalization** by scattering multiple distractor objects across the workspace to create visual interference; and **(3) Position Generalization** by placing the target object in a spatial position outside the training distribution. As shown in Table VI, LaST$_{1.0}$ achieves the highest base success rate (70.0%) and remains robust across all OOD settings. LaST$_{1.0}$ drops by 21% under object generalization and exhibits no performance drop (0%) under background generalization, whereas Cosmos Policy drops by 64% and 36%, respectively. Under the more challenging position generalization, LaST$_{1.0}$ retains a 50.0% success rate (29% drop), while baselines suffer significant drops of over 40%. Overall, across these three

settings, $\text{LaST}_{1.0}$ achieves a 58.3% average success rate, outperforming LaST_0 (41.7%) and Cosmos Policy (21.7%). These results indicate that geometry-aware latent targets and an encoder-free interface improve robustness to changes in object, background, and spatial configuration.

V. CONCLUSION

In this paper, we presented $\text{LaST}_{1.0}$, a unified latent World-Action Model that reformulates predictive world modeling for manipulation as control-centric future prediction in latent space. Rather than modeling an observation-complete video future, $\text{LaST}_{1.0}$ predicts compact, geometry-aware latents focused on future robot–object interactions, providing predictive supervision relevant for control while maintaining action-only deployment. To support this paradigm, we redesigned the visual input interface and attention mechanism with a vision-encoder-free architecture and LaST-Attention for structured spatiotemporal interactions across modalities. Extensive evaluations across simulation and real-world settings demonstrate state-of-the-art performance and strong generalization, validating control-centric latent world modeling as an effective approach for robust manipulation.

REFERENCES

[1] P. Intelligence, K. Black, N. Brown *et al.*, "$\pi_{0.5}$: a vision-language-action model with open-world generalization," 2025.

[2] J. Liu, H. Chen, P. An *et al.*, "Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model," *arXiv preprint arXiv:2503.10631*, 2025.

[3] M. J. Kim, C. Finn, and P. Liang, "Fine-tuning vision-language-action models: Optimizing speed and success," *arXiv preprint arXiv:2502.19645*, 2025.

[4] Z. Liu, J. Liu, H. Chen *et al.*, "Last _{0}: Latent spatio-temporal chain-of-thought for robotic vision-language-action model," *arXiv preprint arXiv:2601.05248*, 2026.

[5] A. Ye, B. Wang, C. Ni *et al.*, "Gigaworld-policy: An efficient action-centered world–action model," *arXiv preprint arXiv:2603.17240*, 2026.

[6] H. Luo, W. Zhang, Y. Feng *et al.*, "Being-h0. 7: A latent world-action model from egocentric videos," *arXiv preprint arXiv:2605.00078*, 2026.

[7] S. Ye, Y. Ge, K. Zheng *et al.*, "World action models are zero-shot policies," *arXiv preprint arXiv:2602.15922*, 2026.

[8] C.-L. Cheang, G. Chen, Y. Jing *et al.*, "Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation," *arXiv preprint arXiv:2410.06158*, 2024.

[9] M. J. Kim, Y. Gao, T.-Y. Lin *et al.*, "Cosmos policy: Fine-tuning video models for visuomotor control and planning," *arXiv preprint arXiv:2601.16163*, 2026.

[10] G. Team, A. Ye, A. Ma *et al.*, "Gigaworld-policy-0.5: A faster and stronger wam empowered by autoresearch," *arXiv preprint arXiv:2607.13960*, 2026.

[11] T. Yuan, Z. Dong, Y. Liu *et al.*, "Fast-wam: Do world action models need test-time future imagination?" *arXiv preprint arXiv:2603.16666*, 2026.

[12] J. Wang, M. Chen, S. Zhang *et al.*, "Vggt-Ω," *arXiv preprint arXiv:2605.15195*, 2026.

[13] Y. Luo, H. Chen, Z. Wu *et al.*, "Look before acting: Enhancing vision foundation representations for vision-language-action models," *arXiv preprint arXiv:2603.15618*, 2026.

[14] J. Wang, Q. Zhang, S. Yang *et al.*, "Repwam: World action modeling with representation visual-action tokenizers," *arXiv preprint arXiv:2606.13674*, 2026.

[15] H. Chen, J. Liu, C. Gu *et al.*, "Fast-in-slow: A dual-system foundation model unifying fast manipulation within slow reasoning," *arXiv preprint arXiv:2506.01953*, 2025.

[16] S. Li, Y. Gao, D. Sadigh *et al.*, "Unified video action model," *arXiv preprint arXiv:2503.00200*, 2025.

[17] H. Bi, H. Tan, S. Xie *et al.*, "Motus: A unified latent action world model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026, pp. 35 101–35 113.

[18] J. Sun, W. Zhang, Z. Qi *et al.*, "Vla-jepa: Enhancing vision-language-action model with latent world model," *arXiv preprint arXiv:2602.10098*, 2026.

[19] H. Chen, J. Liu, Z. Yan *et al.*, "Last-r1: Reinforcing robotic manipulation via adaptive physical latent reasoning," *arXiv preprint arXiv:2604.28192*, 2026.

[20] T. Yuan, Y. Liu, C. Lu, Z. Chen, T. Jiang, and H. Zhao, "Depthvla: Enhancing vision-language-action models with depth-aware spatial reasoning," *arXiv preprint arXiv:2510.13375*, 2025.

[21] D. Qu, H. Song, Q. Chen *et al.*, "Spatialvla: Exploring spatial representations for visual-language-action model," *arXiv preprint arXiv:2501.15830*, 2025.

[22] Z. Liu, J. Liu, J. Xu *et al.*, "Mla: A multisensory language-action model for multimodal understanding and forecasting in robotic manipulation," *arXiv preprint arXiv:2509.26642*, 2025.

[23] C. Li, J. Wen, Y. Peng, Y. Peng, and Y. Zhu, "Pointvla: Injecting the 3d world into vision-language-action models," *IEEE Robotics and Automation Letters*, vol. 11, no. 3, pp. 2506–2513, 2026.

[24] Y. Jia, J. Liu, S. Chen *et al.*, "Lift3d policy: Lifting 2d foundation models for robust 3d robotic manipulation," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 17 347–17 358.

[25] X. Li, L. Heng, J. Liu *et al.*, "3ds-vla: A 3d spatial-aware vision language action model for robust multi-task manipulation," in *9th Annual Conference on Robot Learning*, 2025.

[26] F. Li, W. Song, H. Zhao, J. Wang, P. Ding, D. Wang, L. Zeng, and H. Li, "Spatial forcing: Implicit spatial representation alignment for vision-language-action model," in *International Conference on Learning Representations*, vol. 2026, 2026, pp. 132 324–132 345.

[27] T. Lin, G. Li, Y. Zhong, Y. Zou, Y. Du, J. Liu, E. Gu, and B. Zhao, "Evo-0: Vision-language-action model with implicit spatial understanding," *arXiv preprint arXiv:2507.00416*, 2025.

[28] A. Abouzeid, M. Mansour, Q. Sun *et al.*, "Geoaware-vla: Implicit geometry aware vision-language-action model," *arXiv preprint arXiv:2509.14117*, 2025.

[29] L. Yang, W. Song, X. Wang *et al.*, "4d-wam: Infusing spatiotemporal awareness into world action models through trajectory fields," *arXiv preprint arXiv:2608.08023*, 2026.

[30] A. Yang, A. Li, B. Yang *et al.*, "Qwen3 technical report," *arXiv preprint arXiv:2505.09388*, 2025.

[31] N. Kachaev, M. Kolosov, D. Zelezetsky *et al.*, "Don't blind your vla: Aligning visual representations for ood generalization," *arXiv preprint arXiv:2510.25616*, 2025.

[32] Y. Lipman, R. T. Chen, H. Ben-Hamu *et al.*, "Flow matching for generative modeling," *arXiv preprint arXiv:2210.02747*, 2022.

[33] J. Ye, N. Gao, S. Yang *et al.*, "Starvla-α: Reducing complexity in vision-language-action systems," *arXiv preprint arXiv:2604.11757*, 2026.

[34] L. Li, Q. Zhang, Y. Luo *et al.*, "Causal world modeling for robot control," *arXiv preprint arXiv:2601.21998*, 2026.

[35] S. James, Z. Ma, D. R. Arrojo *et al.*, "Rlbench: The robot learning benchmark & learning environment," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3019–3026, 2020.

[36] B. Liu, Y. Zhu, C. Gao *et al.*, "Libero: Benchmarking knowledge transfer for lifelong robot learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 44 776–44 791, 2023.

[37] S. Fei, S. Wang, J. Shi *et al.*, "Libero-plus: In-depth robustness analysis of vision-language-action models," *arXiv preprint arXiv:2510.13626*, 2025.

[38] J. Zhou, J. Wang, B. Ma *et al.*, "Uni3d: Exploring unified 3d representation at scale," in *International Conference on Learning Representations*, vol. 2024, 2024, pp. 46 766–46 782.

[39] T. Wan, A. Wang, B. Ai *et al.*, "Wan: Open and advanced large-scale video generative models," *arXiv preprint arXiv:2503.20314*, 2025.

[40] O. Siméoni, H. V. Vo, M. Seitzer, F. Baldassarre, M. Oquab, C. Jose, V. Khalidov, M. Szafraniec, S. Yi, M. Ramamonjisoa *et al.*, "Dinov3," *arXiv preprint arXiv:2508.10104*, 2025.

[41] M. Tschannen, A. Gritsenko, X. Wang *et al.*, "Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features," *arXiv preprint arXiv:2502.14786*, 2025.