2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

# Seeing Space and Motion: Enhancing Latent Actions with Geometric and Dynamic Awareness for Vision-Language-Action Models

Zhejia Cai[1,2,*], Yandan Yang[1], Xinyuan Chang[1], Shiyi Liang[1,3,*],
Ronghan Chen[1], Feng Xiong[1,‡], Mu Xu[1], Ruqi Huang[2,†]

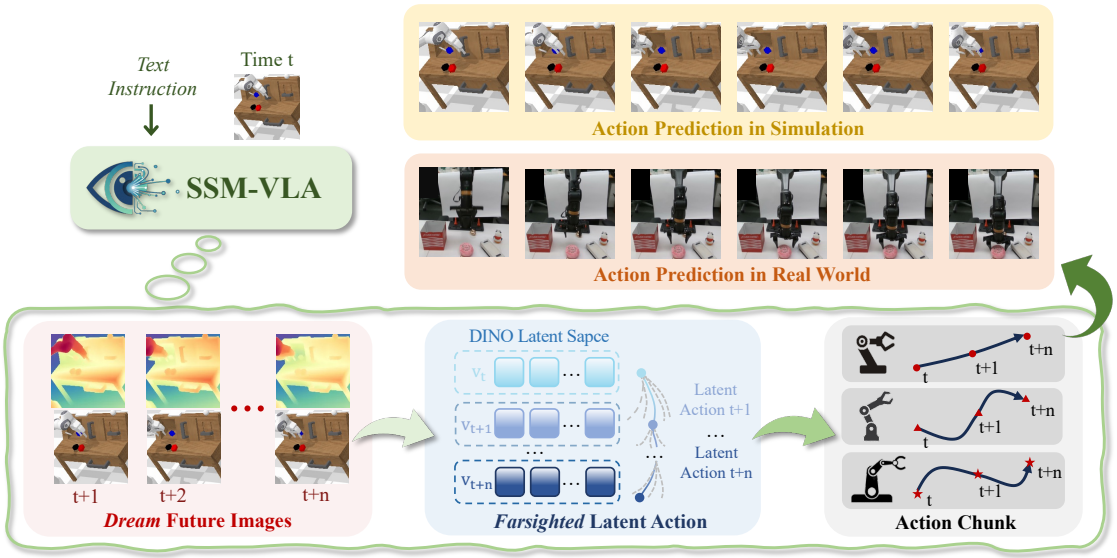


Fig. 1: **Illustration of the end-to-end causal reasoning pipeline within a single SSM-VLA model**, comprising three stages: 1) Future observation prediction for interpretable, temporally coherent reasoning; 2) Farsighted latent action modeling for long-horizon policy planning; 3) Modular action chunk prediction for cross-platform generalization. Experiments in real and simulated environments demonstrate SSM-VLA's effectiveness.

***Abstract*— Latent Action Models (LAMs) enable Vision-Language-Action (VLA) systems to learn semantic action representations from large-scale unannotated data. Yet, we identify two bottlenecks of LAMs: 1) the commonly adopted end-to-end trained image encoder suffers from poor spatial understanding; 2) LAMs can be fragile when input frames are temporally distant, leading to limited temporal perception. Such factors inevitably hinder stable and clear action modeling. To this end, we propose Farsighted-LAM, a latent action framework with geometry-aware spatial encoding and multi-scale temporal modeling, capturing structural priors and dynamic motion patterns from consecutive frames. We further propose SSM-VLA, an end-to-end VLA framework built upon Farsighted-LAM, which integrates structured perception with a visual Chain-of-Thought module to explicitly reason about environmental dynamics, enhancing decision consistency and interpretability. We validate SSM-VLA on multiple VLA tasks in both simulation and real-world settings, and achieve state-of-the-art performance. Our results demonstrate that our strategy of combining geometry-aware modeling, temporal coherence, and explicit reasoning is effective in enhancing the robustness and generalizability of embodied intelligence.**

[1]AMAP, Alibaba Group.
[2]Tsinghua Shenzhen International Graduate School, Tsinghua University.
[3]School of Software Engineering, Xi'an Jiaotong University.
[*]This work was conducted during the internship at Alibaba Group.
[†]Corresponding author: ruqihuang@sz.tsinghua.edu.cn
[‡]Project leader.

## I. Introduction

Latent Action Models (LAMs) have emerged as a promising paradigm for Vision-Language-Action (VLA) systems, enabling self-supervised learning of compact, semantic action representations from large-scale, unannotated vision-language data. By capturing spatial configurations and motion patterns, LAMs facilitate action consequence reasoning and future state anticipation without requiring explicit embodiment or fine-grained action labels. Such properties allow robots to acquire generalizable policies from internet-scale data with minimal real-world interaction. This shift toward scalable, data-driven learning paves the way for more adaptable and generalist agents.

However, existing LAMs remain limited in robust embodied reasoning due to two critical shortcomings: 1) Inadequate spatial understanding, where direct RGB encoding biases latent actions toward surface textures, neglecting geometric structure such as object relations and scene layout; and

2) Limited temporal perception, as most methods rely on sparse, two-frame inputs (e.g., UniVLA [1], Moto-GPT [2]), failing to capture both long-term dynamics and fine-grained motion transitions. This dual deficiency leads to unstable and semantically ambiguous action representations, undermining decision reliability.

To address the above issues, we propose Farsighted-LAM, a latent action modeling framework that enhances spatial and temporal fidelity through two key designs: 1) Geometry-aware spatial encoding using DINOv2 [3] features, which encode structural priors (e.g., spatial layouts, implicit depth, and object relations) with geometrically consistent and semantics-rich scene understanding; and 2) Multi-scale temporal modeling via consecutive frame sequences, capturing both sustained motion trends and transient interactions (e.g., contacts, manipulations), thereby improving temporal coherence and prediction stability. Together, these advances enable more structured and dynamic environment modeling.

Building upon Farsighted-LAM, we further introduce **S**eeing **S**pace and **M**otion (**SSM**)**-VLA**, an end-to-end VLA framework that integrates structured perception with a Chain-of-Thought (CoT) reasoning module to explicitly simulate environmental dynamics before action execution, enhancing interpretability and physical plausibility. We validate SSM-VLA in both simulation and real-world robotic tasks, achieving state-of-the-art performance on CALVIN ABC-D benchmark. Our results demonstrate that our strategy is effective in enhancing the robustness and generalizability of embodied intelligence.

In summary, our contributions are as follows:

- We propose Farsighted-LAM, a latent action model with enhanced spatial understanding and multi-scale temporal modeling, enabling robust representation of scene structure and dynamic motion patterns.
- We propose SSM-VLA, an end-to-end VLA framework that integrates Farsighted-LAM for geometry-aware spatiotemporal modeling with a visual CoT module, enhancing decision consistency and interpretability.
- Through comparisons with competitive baseline models, we show that SSM-VLA achieves state-of-the-art performance on a challenging VLA benchmark.

## II. Related Works

### A. Vision-Language-Action Models

A dominant paradigm in robot learning trains end-to-end policies that directly map high-dimensional sensory inputs to low-level actions. Pioneered by generalist agents like Gato [4] and established in robotics by Octo [5] and RT-1 [6], this approach has been scaled via fine-tuning vision-language models (e.g., RT-2 [7], OpenVLA [8]) and generative methods such as Diffusion Policy [9] and its transformer-based variants [10], [11], [12]. The strength of this model lies in integrating perception, reasoning, and control into a single framework. However, this direct end-to-end prediction method still faces three fundamental challenges: insufficient support from observational information for action decision-making, over-coupling of policy learning with physical carrier characteristics, and difficulty in effectively utilizing unlabeled video data rich in physical and interaction dynamics.

### B. Latent Action Pretraining

To bridge the gap between representation learning and control, prior work pre-trains visual representations from video (e.g., Genie [13], Dynamo [14], R3M [15]) for downstream adaptation. Building on this, Latent Action Models (LAMs) learn robot-independent latent actions from observation pairs through an inverse dynamics model, captures the intent of state transitions, and uses a lightweight network to decode them into embodied motor commands. LAMs have evolved from early frameworks like IGOR [16] and LAPO [17] to large-scale unsupervised pre-training with vision-language models such as LAPA [18], reducing reliance on labeled data. Extensions like UniAct [19] and UniVLA [1] enable cross-embodiment generalization via universal or task-centric latent actions, achieving state-of-the-art efficiency. VideoWorld [20] models multi-frame images but focuses on knowledge acquisition instead of CoT reasoning in Vision-Language-Action models. Our method, Farsighted-LAM, addresses this issue and significantly enhances the model's geometric cognition and dynamic awareness.

### C. Visual Expectation Enhancement

Existing work, known as Inverse Dynamics Models (IDM), focuses on using visual expectations to enhance the context of the current observation for action prediction. VideoAgent [21] is a self-improving system that refines generated video plans for robot control by leveraging environmental feedback to correct hallucinations and boost task success. Gen2Act [22] enables robots to generalize to novel tasks by conditioning a single policy on generated human videos. Seer [23] introduces an end-to-end paradigm that learns scalable robot policies by predicting actions from its own forecasted visual states. The Video Prediction Policy (VPP) [24] achieves state-of-the-art robotic manipulation by conditioning its actions on the rich, "predictive visual representations" extracted from pre-trained Video Diffusion Models. Our method further enhances action guidance by forecasting future visual states imbued with geometric priors, leading to more precise control in 3D space.

## III. Method

In this section, we introduce the details of our proposed SSM-VLA. First, we design Farsighted Latent Action Model to learn latent action with enhancement of dynamic spatial information in III-A. Then we introduce the overall VLA policy in III-B with three stages of prediction: VisualCoT, latent action, and action prediction.

### A. Farsighted Latent Action Model

Latent action model(LAM) learns a structured latent action space from unlabeled videos, aiming to improve the

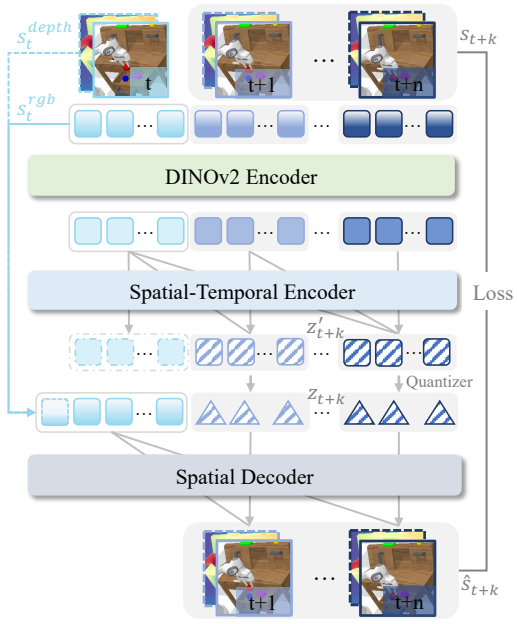


Fig. 2: **Architecture of our Farsighted Latent Action Model.** The encoder takes DINOv2 features of the current frame $s_t$ and multiple future keyframes to predict a sequence of latent actions. The decoder then uses the current frame $s_t$ and a quantized latent action $z_{t+k}$ to reconstruct the corresponding future frame $\hat{s}_{t+k}$.

generalization ability of VLA systems. It usually takes in a pair of observations, marked as $s_t, s_{t+K}$, to compute the latent action $a_t$ at time $t$. Here $K$ is a fixed time interval between these two observations. Here we modify this process and design this model to enhance both spatial reasoning and dynamic modeling ability for the latent action, as illustrated in Fig. 2.

*1)* ***Encoder****:* In contrast to conventional LAMs operating on a pair of observations, our model extends the receptive field of LAM and simultaneously processing a sequence of future $N$ key frames $\{s_{t+i}\}_{i=1}^{N}$ to predict a corresponding sequence of latent actions in a single forward pass. This helps the model capture both detailed and long-horizon motion information of the dynamic scene. Meanwhile, to reason with deeper awareness of the structural priors in spatial domain, we take not only RGB image but also depth for the observation $s_t$, which is denoted as $s_t = (s_t^{rgb}, s_t^{depth})$. RGB image is used as input and depth is utilized for additional supervision. We further leverage features $v_t$ from a frozen DINOv2 encoder, denoted as $\Phi_V$ to ground the representation in a geometrically and semantically rich space. Specifically, we first extract the corresponding visual feature for each RGB frame, where $v_t = \Phi_V(s_t^{rgb})$, and get a sequence of feature $\{v_{t+i}\}_{i=0}^{N}$.

These features, along with a set of $N$ learnable *latent action queries*, $\mathcal{Q} = \{q_k\}_{k=1}^{N}$, are processed by a spatio-temporal transformer $\Psi_{\text{ST}}$ to jointly encode the space and motion. For each query $q_k \in \mathcal{Q}$, the transformer generates a sequence of continuous latent vectors, each representing a continuous latent action for the future timestep $t+k$:

$$z'_{t+k} = \Psi_{\text{ST}}(\{v_{t+i}\}_{i=0}^{k}, q_k), \tag{1}$$

where $k \in \{1, 2, ..., N\}$. Subsequently, each continuous latent vector $z'_{t+k}$ is quantized to a discrete token $z_{t+k}$ via a nearest-neighbor lookup in a learned codebook $\mathcal{C} \subset \mathbb{R}^D$:

$$z_{t+k} = \arg\min_{c \in \mathcal{C}} \|z'_{t+k} - c\|_2 \tag{2}$$

The final encoded feature is a sequence of discrete tokens $\{z_{t+k}\}_{k=1}^{N}$, which constitutes the quantized representation of the future action plan and serves as the latent action.

*2)* ***Decoder****:* The decoder is responsible for validating the semantic and dynamic content of a learned latent action by translating it back into the visual domain. Here we implement the decoder as a spatial transformer, denoted as $\Psi_{\text{S}}$. It predicts the future observation $\hat{s}_{t+k}$ at time $t+k$ given only the initial observation $s_t$ and a discrete latent action $z_{t+k}$ of the future time step $t+k$. We emphasize that the predicted observation $\hat{s}_{t+k}$ includes both RGB $\hat{s}_{t+k}^{rgb}$ and depth $\hat{s}_{t+k}^{depth}$. This makes sure that the latent action has learned not only the dynamic information of visual texture but also the spatial structure of the scene. The generation process is formulated as:

$$\hat{s}_{t+k} = (\hat{s}_{t+k}^{rgb}, \hat{s}_{t+k}^{depth}) = \Psi_{\text{S}}(s_t^{rgb}, s_t^{depth}, z_{t+k}) \tag{3}$$

Different from the encoder, here we restrict the input of the decoder to the tuple of $(s_t^{rgb}, z_{t+k})$, making the decoder blind to the ground-truth target observation $s_{t+k}$ or any intermediate observations $\{s_{t+j}\}_{j=1}^{k-1}$. This constraint prevents the decoder from learning shortcut mappings with adjacent frames, which conversely enforces the encoder to embed more space and motion information into the latent action $z_{t+k}$. We count on the latent action $z_{t+k}$ to bridge the gap from $s_t$ to $s_{t+k}$ in the decoding process as well as the overall VLA policy in Sec. III-B.

*3)* ***Reconstruction Loss****:* We propose a multi-modal reconstruction loss $\mathcal{L}_{\text{rec}}$ to supervise the farsighted latent action model. Since the decoder generates both RGB $\hat{s}_{t+k}^{rgb}$ and depth $\hat{s}_{t+k}^{depth}$, we conduct loss on these two modal predictions according to the ground-truth observations $s_{t+k}^{rgb}$ and $s_{t+k}^{depth}$.

The photometric loss $\mathcal{L}_{\text{rgb}}$ combines $L2$ loss with the LPIPS perceptual loss [25] with weight $\lambda_{LPIPS}$:

$$\mathcal{L}_{\text{rgb}}(s, \hat{s}) = \|\hat{s}^{\text{rgb}} - s^{\text{rgb}}\|_2^2 + \lambda_{\text{lpips}} \cdot \mathcal{L}_{\text{lpips}}(\hat{s}^{\text{rgb}}, s^{\text{rgb}}) \tag{4}$$

This constraint primarily ensures that the rendered output is photorealistic and captures the correct appearance, which is crucial for strengthening the model's understanding of semantic content like textures and object identities.

Then the depth loss $\mathcal{L}_{\text{depth}}$ is a gradient-aware logarithmic loss [26] that inversely weights the loss by the RGB image gradient:

$$\mathcal{L}_{\text{depth}}(s, \hat{s}) = \exp(-\|\nabla s^{\text{rgb}}\|) \cdot \frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \log(1 + |\hat{s}^{\text{depth}}(p) - s^{\text{depth}}(p)|), \tag{5}$$

where $p \in \mathcal{P}$ means each pixel of the depth map. This depth constraint enforces geometric consistency, which is fundamental to a robust understanding of the underlying 3D spatial structure.

The final reconstruction loss is a weighted sum over all $N$ future key frames. By integrating the photometric loss $\mathcal{L}_{\text{rgb}}$ and the depth loss $\mathcal{L}_{\text{depth}}$, we ensure that the model learns a representation that is faithful in both appearance and geometry. The hyperparameter $\lambda_{\text{d}}$ balances the relative importance of these two constraints:

$$\mathcal{L}_{\text{rec}} = \sum_{k=1}^{N} \left( \mathcal{L}_{\text{rgb}}(s_{t+k}, \hat{s}_{t+k}) + \lambda_{\text{d}} \mathcal{L}_{\text{depth}}(s_{t+k}, \hat{s}_{t+k}) \right). \quad (6)$$

### B. The Overall VLA Policy

As shown in Fig. 3, the SSM-VLA model takes the current visual observation and natural language commands as input. Inspired by Moto-GPT [2], the model first predicts foresightful implicit actions as intermediate representations using a Foresight Implicit Action Model (F-LAM). These implicit actions remain semantically abstract and task-generalizable, effectively decoupling high-level task intent from low-level execution details. Subsequently, an action query mechanism combined with a diffusion-based policy adapts the implicit actions to the target robot's action space.

Furthermore, we observe a significant modal gap in the direct mapping, which can easily lead to optimization instability and decreased generalization ability. Inspired by FSDrive [27], we introduce a Visual CoT mechanism, using the prediction of future visual states (e.g., RGB or depth frames) as an intermediate inference step. This "imagine first, then act" paradigm both strengthens the model's spatiotemporal understanding and improves the accuracy and temporal coherence of generated actions significantly.

In summary, our model operates in three cascaded stages: VisualCoT Prediction, Farsighted Latent Action Inference, and Action Generation. During fine-tuning, each stage takes a query vector as additional input and is supervised by a specific objective. We detail the implementation of each stage here.

*1)* ***Stage 1: VisualCoT Prediction****:* A visual prediction module, $\mathcal{M}_{\text{vision}}$, takes the historical observations $s_{t-H:t}$ ($H$ is the history length) and language instruction $l$ to generate the next visual state:

$$\hat{s}_{t+1} = \mathcal{M}_{\text{vision}}(s_{t-H:t}, l) \quad (7)$$

This prediction is supervised by a vision loss, $\mathcal{L}_{\text{vision}}$. We adopt the same loss formulation as for reconstruction (see Eq. (6)), supervising both the predicted RGB image and depth map. This enables the VLA model to concurrently forecast both future semantic observations (via RGB) and geometric structure (via depth). For data with sensor depth, it is formally identical to the reconstruction loss $\mathcal{L}_{\text{rec}}$ applied to the single next frame:

$$\mathcal{L}_{\text{vision}} = \mathcal{L}_{\text{rec}}(s_{t+1}, \hat{s}_{t+1}) \quad (8)$$

For data without sensor depth, since depth maps predicted from a single image by methods like DepthAnything [28] are inherently normalized and lack a metric scale, it is necessary to align these varied predictions into a consistent world coordinate system. To this end, we follow [26] and generate a pseudo-target. An initial estimate $D_{\text{mono}}$ is aligned to a sparse map $D_{\text{sparse}}$, obtained via SfM [29] using different views and poses of camera. The alignment is solved via closed-form linear regression over the set of sparse pixels $\mathcal{P}_{\text{sparse}}$:

$$\hat{a}, \hat{b} = \arg\min_{a,b} \sum_{p \in \mathcal{P}_{\text{sparse}}} \| (a \cdot D_{\text{mono}}(p) + b) - D_{\text{sparse}}(p) \|_2^2 \quad (9)$$

Then we get the pseudo target of depth with $s^{depth} = \hat{a} \cdot D_{\text{mono}} + \hat{b}$ and apply the visual loss in Eq. (8) on it

*2)* ***Stage 2: Farsighted Latent Action Inference****:* The latent prediction module, $\mathcal{M}_{\text{latent}}$, takes the historical context and the predicted next frame's features to infer a sequence of future action-intent distributions with length $N$:

$$\hat{z}_{t+k} = \mathcal{M}_{\text{latent}}(s_{t-H:t}, l, \hat{s}_{t+1}, \{\hat{z}_{t+j}\}_{j=1}^{k-1}), \quad (10)$$

where $k \in \{1, 2, ..., N\}$. It is supervised by ground-truth latent action $z_{t+k}$, which is generated by the fixed encoder of the previously trained Farsighted Latent Action Model (F-LAM). F-LAM here takes in the ground-truth video frames. Note here the predicted $\hat{z}_{t+k}$ has been projected to the space of discrete latent action $z_{t+k}$. Thus, here we use a Cross-Entropy loss for the latent action:

$$\mathcal{L}_{\text{latent}} = -\sum_{k=1}^{N} z_{t+k} \log(\hat{z}_{t+k}) \quad (11)$$

*3)* ***Stage 3: Action Generation****:* Then the action module, $\mathcal{M}_{\text{action}}$, takes a comprehensive context vector, including the historical context and the predicted latent action to generate the intermediate feature $c_t$ of the robot action:

$$c_t = \mathcal{M}_{\text{action}}(s_{t-H:t}, l, \hat{s}_{t+1}, \{\hat{z}_{t+j}\}_{j=1}^{N}) \quad (12)$$

Then we utilize this feature $c_t$ as the condition of a conditional Flow Matching model $V_\theta$ [30] to predict the real action. The corresponding loss is:

$$\mathcal{L}_{\text{action}} = \mathbb{E}_{\tau,\epsilon,a_t} \left[ \| V_\theta(\tau a_t + (1-\tau)\epsilon, \tau, c_t) - (\epsilon - a_t) \|_2^2 \right] \quad (13)$$

where $V_\theta$ is a DiT network and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

The entire VLA policy is fine-tuned by minimizing a composite loss, $\mathcal{L}_{\text{VLA}}$, which integrates the objectives from each training stage through a weighted summation:

$$\mathcal{L}_{\text{VLA}} = \mathcal{L}_{\text{action}} + \lambda_{\text{latent}} \mathcal{L}_{\text{latent}} + \lambda_{\text{vision}} \mathcal{L}_{\text{vision}} \quad (14)$$

### C. Multi-modal Synergistic Attention

The cascaded architecture of SSM-VLA is implemented within a single, unified transformer through a carefully designed Multi-modal Synergistic Attention mechanism. Initially, historical visual tokens $(s_{t-H:t})$ and language tokens $(l)$ form a bi-directionally attentive core context, grounding the instruction in visual history. Subsequently, the visual prediction stage queries attend only to this core context to

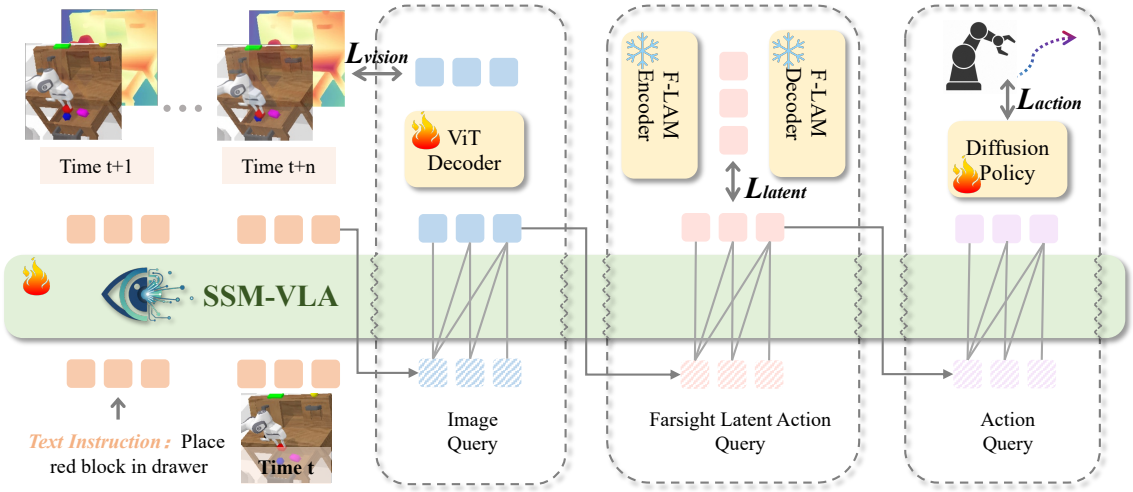


Fig. 3: **The Three-Stage Cascaded VLA Policy.** Stage 1 predicts the immediate future observation $\hat{s}_{t+k}$. Stage 2 infers a long-horizon latent action plan $\{\hat{z}_{t+k}\}_{k=1}^{N}$. Stage 3 fuses all information to produce the final executable action $a_t$.

generate $\hat{s}_{t+1}$, ensuring the prediction is strictly a function of the past. The latent planning stage then takes a step further, with its queries attending to both the core context and the predicted frame $\hat{s}_{t+1}$ to produce the plan $\{\hat{z}_{t+j}\}_{j=1}^{N}$; a causal mask within these queries ensures the plan's temporal coherence. Finally, the action query $a_t$ acts as the final information aggregator, focusing on the core context, the predicted next frame, and the complete implicit plan. This progressively structured attention mechanism enables each component to specialize based on its preceding output, synergistically driving the reasoning capabilities of SSM-VLA.

## IV. Evaluations

### A. Implementation Details

All models are implemented in PyTorch. Both models employ a cosine learning rate schedule with a 5% linear warm-up phase. For the Latent Action Model, we use an AdamW optimizer [35] with an initial learning rate of $10^{-4}$ and a weight decay of $10^{-5}$. The batch size is set to 256 while total training step is set to 100, and the size of codebook $\mathcal{C}$ is set to 32. For the CALVIN benchmark, this model is trained to encode the latent action between the current frame and the subsequent 3 frames, while for real-world data, it models the action between the current frame and the next 2 frames. Each frame is represented by 4 discrete tokens, and the flow matching head uses 10 denoising steps. The loss weights for this model are set as $\lambda_{\text{lpips}} = 1$, $\lambda_{rgb} = 1$, and $\lambda_d = 0.01$. For the VLA Model, we also use an AdamW optimizer, but with an initial learning rate of $10^{-3}$ and a weight decay of $10^{-4}$. The batch size for this model is 64 while total training step is set to 30. Its loss weights are set as $\lambda_{\text{vision}} = 0.1$ and $\lambda_{\text{latent}} = 0.01$.

### B. Simulation Benchmark Experiments

*1) Experiments Setup:* We evaluate our approach on the CALVIN [36] benchmark, which consists of 34 distinct manipulation tasks defined by open-ended language instructions, ranging from simple pick-and-place to complex articulated object manipulation. The benchmark utilizes a Franka Panda robotic arm across four tabletop environments. For our experiments, we trained policies on demonstration data from environments A, B, and C, and then undergo zero-shot evaluation in the unseen environment D. We utilized no-language-instruction data to pretrain our model following methods in Seer [23]. The policy's generalization is rigorously tested on 1,000 unique instruction chains, each requiring the completion of five consecutive tasks. We utilized all of three types of signals (static camera, gripper camera and proprioceptive state) as input to achieve best performance. As presented in Table 1, the results on this demanding CALVIN ABC-D benchmark show that SSM-VLA achieves top performance.

*2) Results:* The results, as presented in Table 1, show that SSM-VLA achieves top performance on the ABC-D benchmark, outperforming a wide array of existing methods. Specifically, SSM-VLA surpasses direct prediction models (e.g., Roboflamingo [10], Dita [11]), latent-to-real action models (e.g., Moto-GPT [2], UniVLA [1]), and integrated visual foresight models (e.g., Seer [23], VPP [24]). We attribute this success to our model's unique cascaded architecture, which enables superior multi-task learning and generalization. Figure 4 presents visualizations for three distinct scenarios. For each scenario, we show the results from five simulations, capturing a snapshot every five actions. Collectively, these results demonstrate our model's effectiveness in a multi-task learning setting.

### C. Real World Experiments

We evaluate our approach on a real-world robotic manipulation task using a single AgileX Piper robot, tasked with placing a pink toy into a box. The model is first pretrained on the large-scale Open-X-Embodiment dataset [37], [38], [39], and subsequently fine-tuned on 50 human-collected demon-

TABLE I: Comparative evaluation on the CALVIN benchmark. Our method demonstrates state-of-the-art performance, surpassing all baselines with higher success rates for N-length task sequences and a greater average successful chain length.

| **Method** | **Task completed in a row** | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | **Avg. Len. ↑** |
| Roboflamingo (ICLR 24) [10] | 82.4 | 61.9 | 46.6 | 33.1 | 23.5 | 2.47 |
| Susie (ICLR 24) [31] | 87.0 | 69.0 | 49.0 | 38.0 | 26.0 | 2.69 |
| Moto-GPT (ICCV 25) [2] | 89.7 | 72.9 | 60.1 | 48.4 | 38.6 | 3.10 |
| 3D Diffusor Actor (CoRL 25) [12] | 93.8 | 80.3 | 66.2 | 53.3 | 41.2 | 3.35 |
| CLOVER (NeurIPS 24) [32] | 96.0 | 83.5 | 70.8 | 57.5 | 45.4 | 3.53 |
| Dita (ICCV 25) [11] | 94.5 | 82.5 | 72.8 | 61.3 | 50.0 | 3.61 |
| RoboDual (CoRR 24) [33] | 94.4 | 82.7 | 72.1 | 62.4 | 54.4 | 3.66 |
| UniVLA (RSS 25) [1] | 95.5 | 85.8 | 75.4 | 66.9 | 56.5 | 3.80 |
| UP-VLA (ICML 25) [34] | 92.8 | 86.5 | 81.5 | 76.9 | 69.9 | 4.08 |
| Seer (ICLR 25) [23] | 96.3 | 91.6 | 86.1 | 80.3 | 74.0 | 4.28 |
| VPP (ICML 25) [24] | 95.7 | 91.2 | 86.3 | 81.0 | 75.0 | 4.29 |
| **SSM-VLA** | **97.6** | **94.1** | **88.3** | **81.8** | **75.9** | **4.38** |

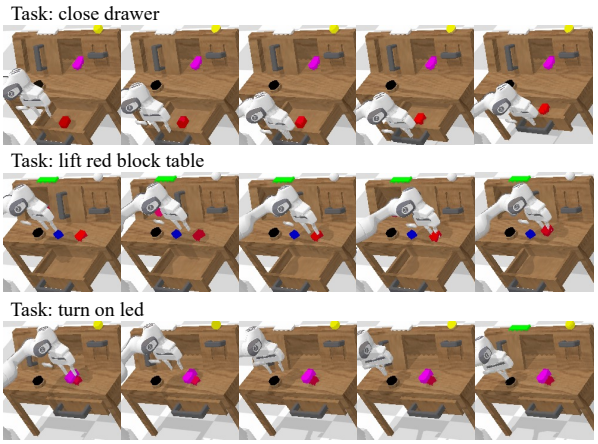


Fig. 4: **Visualization of simulation evaluation tasks.** We visualize the simulation results of three different tasks, which demonstrates success of our model in multi-task learning.

TABLE II: Ablation study of proposed structure.

| Method | **Task completed in a row** | | | | | **Avg. Len. ↑** |
|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | |
| **Full Model** | **97.6** | **94.1** | **88.3** | **81.8** | **75.9** | **4.38** |
| LAM (1-frame) | 96.6 | 92.8 | 86.3 | 79.3 | 72.8 | 4.28 |
| w/o LAM | 96.5 | 92.4 | 85.5 | 78.0 | 70.8 | 4.23 |
| Causal Atten. | 93.8 | 83.7 | 73.3 | 63.4 | 55.6 | 3.70 |
| w/o Depth | 97.3 | 91.3 | 86.0 | 80.7 | 74.4 | 4.30 |

strations. We masked the gripper camera input for real-world experiment because of limited condition. As demonstrated in Figure 5, our method achieves successful deployment on the physical robot and exhibits strong generalization to real-world conditions, including cluttered and unstructured environments.

Fig. 5: **Visualization of the real world experiments.** The model is asked to place the pink ball into the box. We show two samples with different layouts and chaos background.

### *D. Ablation Study*

To verify the benefit of our Farsighted-LAM, Multi-modal Synergistic Attention and Geometric Priors, we conduct an ablation study as shown in Table II. All ablation experiments are conducted on the CALVIN benchmark and run with the same training budget for consistency.

*1) Importance of Farsighted LAM Structure:* Our full model, which uses a 3-frame context (i.e., **Full Model**), achieves the best performance with an average task chain length of 4.38. In comparison, a more "direct" variant using only a single future frame (i.e., **LAM (1-frame)**) reduces the average length to 4.28, while a model where the LAM module is removed entirely (i.e., **w/o LAM**) causes a more significant performance drop to an average length of 4.23. This two-tiered comparison indicates: 1) compared to predicting only a single future frame, using a 3-frame context introduces intermediate-state constraints and provides more continuous dynamic supervision, encouraging the LAM to learn a smoother and more physically consistent latent action representation; 2) removing the LAM entirely significantly reduces the average task chain length, showing that latent action modeling is a key component for long-horizon planning and successful execution. Here we visualize the latent action by decoding it with Farsighted-LAM. As shown in Fig. 6, given the initial frame $s_t^{rgb}$, the first row

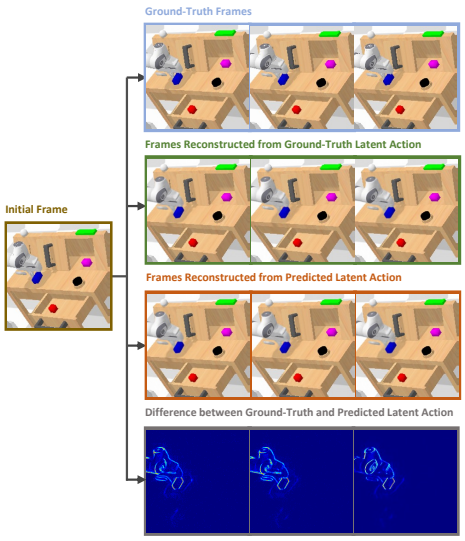


Fig. 6: **Visualization of the latent action.** The similarity of these rows demonstrates that 1) our Farsighted-LAM has learned the dynamic scene with spatial and motion awareness by ground-truth latent action $z_{t+k}$ (i.e., the second row) and 2) the latent action $\hat{z}_{t+k}$ (i.e., the third row) predicted by our VLA model aligns well with the ground-truth latent action.

shows three ground-truth frames in the future. The frames shown in the second row are reconstructed with ground-truth latent actions. Given the pretrained Farsighted-LAM, it first apply the encoder to get the ground-truth latent action $z_t$ and then reconstructed the frame $\hat{s}_{t+k}$ with the decoder. The reconstructed frames precisely track the spatial and motion changes. This confirms that the Farsighted-LAM has learned the dynamic scene with the well-designed structure. In the third row, we reconstruct frames using the same decoder, but instead of ground-truth actions, we feed in the latent action predicted by the overall VLA model as described in Sec. III-B.2. Results show the second and third rows also look similar, which confirms that the latent action $\hat{z}_{t+k}$ predicted by our VLA model aligns well with the ground-truth latent action $z_{t+k}$. We visualize the difference between the second row and the third row in the fourth row.

*2) Effect of Multi-modal Synergistic Attention:* We compared the multi-modal synergistic attention mechanism with a baseline method that uses a simple token-level causal attention mask and both were trained with the same number of steps. As shown in Table II, replacing our structured attention with this naive causal baseline (i.e., **Causal Atten.**) leads to a dramatic performance collapse, with the average sequence length falling from 4.38 to 3.70, which significantly demonstrates the importance of our synergistic design. The simple causal attention mechanism leads to information leakage across different modalities (vision, latent action, real action). In contrast, our structured mechanism allows only necessary components to attend to the information needed from their corresponding modalities, preventing them from "shortcut learning" from other modalities.

TABLE III: Comparison on depth-critical vs. non-depth-critical tasks.

| Method | *push into drawer* | | | *push blue block right* | | |
|---|---|---|---|---|---|---|
| | **Success Count** | **Total Count** | **Success Rate** | **Success Count** | **Total Count** | **Success Rate** |
| Full Model | 102 | 129 | 79.1% | 52 | 71 | 73.2% |
| w/o Depth | 95 | 129 | 73.6% | 51 | 69 | 73.9% |

*3) Contribution of Geometric Priors:* To investigate the contribution of explicit 3D geometric information, we train a policy variant without any depth supervision (i.e., **w/o Depth**) in Table II. Removing depth yields a consistent but moderate degradation across metrics, with the average task chain length decreasing slightly from 4.38 to 4.30. To better characterize when depth helps, we further compare tasks with different degrees of geometric dependency in Table III. On the more depth-critical *push into drawer* task, removing depth supervision reduces the success rate from 79.1% to 73.6%. In contrast, on the largely color-driven *push blue block right* task, the two methods perform similarly. These results suggest that explicit depth supervision mainly benefits tasks which require accurately inferring relative object poses and manipulation affordances.

## V. Limitations and Conclusion

Concurrent works [40], [41] have also highlighted the importance of depth priors. However, how to prevent overfitting during latent action model training remains a highly valuable research direction. In this work, we introduced a novel architecture designed to address the critical limitations of existing Latent Action Models in capturing geometric and dynamic information. By synergistically combining DINOv2 [3] visual features with multi-frame temporal modeling, our model better captures both static scene structure and motion dynamics. The Chain-of-Thought reasoning pipeline further improves prediction by explicitly modeling environmental changes before action selection. Our method achieves state-of-the-art in both simulation and real-world VLA benchmark tests, and highlights the significant value of integrating spatiotemporal cognition into embodied intelligence systems.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China under contract No. 62171256. ChatGPT was used exclusively for improving grammar in Sec I, II, V.

## References

[1] Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li, "Univla: Learning to act anywhere with task-centric latent actions," *arXiv preprint arXiv:2505.06111*, 2025.

[2] Y. Chen, Y. Ge, W. Tang, Y. Li, Y. Ge, M. Ding, Y. Shan, and X. Liu, "Moto: Latent motion token as the bridging language for learning robot manipulation from videos," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 19 752–19 763.

[3] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *Transactions on Machine Learning Research Journal*, 2024.

[4] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-maron, M. Giménez, Y. Sulsky, J. Kay, J. T. Springenberg, *et al.*, "A generalist agent," *Transactions on Machine Learning Research*.

[5] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, L. Y. Chen, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, and S. Levine, "Octo: An open-source generalist robot policy," in *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

[6] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, *et al.*, "Rt-1: Robotics transformer for real-world control at scale," *arXiv preprint arXiv:2212.06817*, 2022.

[7] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Conference on Robot Learning*. PMLR, 2023, pp. 2165–2183.

[8] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. P. Foster, P. R. Sanketi, Q. Vuong, *et al.*, "Openvla: An open-source vision-language-action model," in *Conference on Robot Learning*. PMLR, 2025, pp. 2679–2713.

[9] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, vol. 44, no. 10-11, pp. 1684–1704, 2025.

[10] X. Li, M. Liu, H. Zhang, C. Yu, J. Xu, H. Wu, C. Cheang, Y. Jing, W. Zhang, H. Liu, *et al.*, "Vision-language foundation models as effective robot imitators," in *The Twelfth International Conference on Learning Representations*.

[11] Z. Hou, T. Zhang, Y. Xiong, H. Duan, H. Pu, R. Tong, C. Zhao, X. Zhu, Y. Qiao, J. Dai, *et al.*, "Dita: Scaling diffusion transformer for generalist vision-language-action policy," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 7686–7697.

[12] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki, "3d diffuser actor: Policy diffusion with 3d scene representations," in *Conference on Robot Learning*. PMLR, 2025, pp. 1949–1974.

[13] J. Bruce, M. Dennis, A. Edwards, J. Parker-Holder, Y. Shi, E. Hughes, M. Lai, A. Mavalankar, R. Steigerwald, C. Apps, *et al.*, "Genie: generative interactive environments," in *Proceedings of the 41st International Conference on Machine Learning*, 2024, pp. 4603–4623.

[14] Z. J. Cui, H. Pan, A. Iyer, S. Haldar, and L. Pinto, "Dynamo: In-domain dynamics pretraining for visuo-motor control," *Advances in Neural Information Processing Systems*, vol. 37, pp. 33 933–33 961, 2024.

[15] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 892–909.

[16] X. Chen, J. Guo, T. He, C. Zhang, P. Zhang, D. C. Yang, L. Zhao, and J. Bian, "Igor: Image-goal representations are the atomic control units for foundation models in embodied ai," *arXiv preprint arXiv:2411.00785*, 2024.

[17] X. Chen, A. Ghadirzadeh, T. Yu, J. Wang, A. Y. Gao, W. Li, L. Bin, C. Finn, and C. Zhang, "Lapo: Latent-variable advantage-weighted policy optimization for offline reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 36 902–36 913, 2022.

[18] S. Ye, J. Jang, B. Jeon, S. J. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin, *et al.*, "Latent action pretraining from videos," in *The Thirteenth International Conference on Learning Representations*.

[19] J. Zheng, J. Li, D. Liu, Y. Zheng, Z. Wang, Z. Ou, Y. Liu, J. Liu, Y.-Q. Zhang, and X. Zhan, "Universal actions for enhanced embodied foundation models," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 22 508–22 519.

[20] Z. Ren, Y. Wei, X. Guo, Y. Zhao, B. Kang, J. Feng, and X. Jin, "Videoworld: Exploring knowledge learning from unlabeled videos," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 29 029–29 039.

[21] A. Soni, S. Venkataraman, A. Chandra, S. Fischmeister, P. Liang, B. Dai, and S. Yang, "Videoagent: Self-improving video generation," *arXiv preprint arXiv:2410.10076*, 2024.

[22] H. Bharadhwaj, D. Dwibedi, A. Gupta, S. Tulsiani, C. Doersch, T. Xiao, D. Shah, F. Xia, D. Sadigh, and S. Kirmani, "Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation," in *Conference on Robot Learning*. PMLR, 2025, pp. 3936–3951.

[23] Y. Tian, S. Yang, J. Zeng, P. Wang, D. Lin, H. Dong, and J. Pang, "Predictive inverse dynamics models are scalable learners for robotic manipulation," in *The Thirteenth International Conference on Learning Representations*.

[24] Y. Hu, Y. Guo, P. Wang, X. Chen, Y.-J. Wang, J. Zhang, K. Sreenath, C. Lu, and J. Chen, "Video prediction policy: A generalist robot policy with predictive visual representations," in *International Conference on Machine Learning*. PMLR, 2025, pp. 24 328–24 346.

[25] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[26] M. Turkulainen, X. Ren, I. Melekhov, O. Seiskari, E. Rahtu, and J. Kannala, "Dn-splatter: Depth and normal priors for gaussian splatting and meshing," in *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2025, pp. 2421–2431.

[27] S. Zeng, X. Chang, M. Xie, X. Liu, Y. Bai, Z. Pan, M. Xu, X. Wei, and N. Guo, "Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving," *arXiv preprint arXiv:2505.17685*, 2025.

[28] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth anything v2," *Advances in Neural Information Processing Systems*, vol. 37, pp. 21 875–21 911, 2024.

[29] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4104–4113.

[30] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *The Eleventh International Conference on Learning Representations*.

[31] K. Black, M. Nakamoto, P. Atreya, H. R. Walke, C. Finn, A. Kumar, and S. Levine, "Zero-shot robotic manipulation with pre-trained image-editing diffusion models," in *The Twelfth International Conference on Learning Representations*.

[32] Q. Bu, J. Zeng, L. Chen, Y. Yang, G. Zhou, J. Yan, P. Luo, H. Cui, Y. Ma, and H. Li, "Closed-loop visuomotor control with generative expectation for robotic manipulation," *Advances in Neural Information Processing Systems*, vol. 37, pp. 139 002–139 029, 2024.

[33] Q. Bu, H. Li, L. Chen, J. Cai, J. Zeng, H. Cui, M. Yao, and Y. Qiao, "Towards synergistic, generalized, and efficient dual-system for robotic manipulation," *arXiv preprint arXiv:2410.08001*, 2024.

[34] J. Zhang, Y. Guo, Y. Hu, X. Chen, X. Zhu, and J. Chen, "Up-vla: A unified understanding and prediction model for embodied agent," in *International Conference on Machine Learning*. PMLR, 2025, pp. 74 911–74 922.

[35] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*.

[36] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard, "Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7327–7334, 2022.

[37] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration," in *Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA)*. Institute of Electrical and Electronics Engineers Inc., 2024, pp. 6892–6903.

[38] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, *et al.*, "Droid: A large-scale in-the-wild robot manipulation dataset," in *Robotics: Science and Systems*, 2024.

[39] H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He, V. Myers, M. J. Kim, M. Du, *et al.*, "Bridgedata v2: A dataset for robot learning at scale," in *Conference on Robot Learning*. PMLR, 2023, pp. 1723–1736.

[40] W. Zhang, H. Liu, Z. Qi, Y. Wang, X. Yu, J. Zhang, R. Dong, J. He, F. Lu, H. Wang, *et al.*, "Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge," *arXiv preprint arXiv:2507.04447*, 2025.

[41] 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.