# Latent Action as Intention Enables Efficient Future Imagination for World Action Models

**Xiang Li**[1,2,4*], **Yupeng Zheng**[3*†‡], **Songen Gu**[5*], **Huailiang Ma**[6*], **Feng Yu**[7], **Xian Nie**[7], **Shanshuai Yuan**[4,5], **Yujie Zang**[8], **Weize Li**[8], **Shuai Tian**[3], **Moyang Liu**[9], **Ya-Qin Zhang**[2], **Wenchao Ding**[4†]

[1]CollegeAI, THU [2]AIR, THU [3]CASIA [4]TARS Robotics [5]FDU [6]Southeast University [7]SJTU [8]NUS [9]BUAA

Project page: https://getterupper.github.io/LAWA



**Abstract**

World action models (WAMs) improve robot control by modeling how observations evolve, but generating future observations at test time incurs substantial latency. Fast-WAM removes this process for efficiency; however, our matched implementations show lower generalization for Fast-WAM than for future-aware alternatives, especially with scarce robot demonstrations and in out-of-distribution scenarios. To bridge this gap, we introduce **LAWA**, a WAM architecture that uses compact latent actions as an operational representation of future intentions, enabling efficient test-time future imagination without generating future observations. Specifically, a discrete tokenizer enhanced by action-free pre-training produces manipulation-centric codebook targets. LAWA jointly denoises a continuous latent state anchored to these targets with executable action chunks while omitting the future-video branch at inference. On RoboCasa, LAWA achieves state-of-the-art average success rates of 65.6% and 80.8% in the few-shot and full data settings, improving over the matched Fast-WAM baseline by 9.6 and 4.5 points, respectively. It also preserves the performance level of the matched Joint-WAM variant while requiring 42.9% lower inference latency. LAWA also demonstrates competitive zero-shot robustness on LIBERO-Plus and superior performance on real-world tasks. These results show that future imagination need not be discarded: retaining it with compact latent actions yields an effective trade-off among performance, generalization, and latency. Code and models will be released.

## 1 Introduction

Robotic manipulation requires more than reacting to the current observation: a policy must also understand how an interaction should unfold. World action models (WAMs) address this need by coupling action learning with predictive modeling of future visual dynamics (Liao et al. 2025; Li et al. 2025; Ma et al. 2026; Yu et al. 2026b). Predicting future observations provides the action generator with an explicit representation of task progress and environmental dynamics, which is particularly valuable when labeled robot demonstrations are scarce (Liang et al. 2025b; Kim et al. 2026b). This Joint-WAM mechanism (Bi et al. 2026; Ye et al. 2026b; Li et al. 2026a), however, creates a practical bottleneck. Iteratively denoising future observations adds substantial test-time computation before an action can be executed, thereby posing an obstacle to real-robot control.

Fast-WAM (Yuan et al. 2026b) challenges the need for test-time future imagination: it preserves video prediction during training but removes the future-video branch at inference. Our controlled comparison in Figure 1, however, shows that it generalizes markedly worse than matched Joint-WAM, especially with few-shot supervision. This result suggests that the benefit of future modeling may not arise solely from video co-training and motivates retaining an explicit future representation. Yet retaining observation-space imagination incurs the full cost of visual generation, leaving a trade-off between generalization performance and inference efficiency in current WAM paradigm designs.

We introduce **LAWA**, which addresses this imbalance by moving future imagination from the observation space to a compact latent-action space. Our central idea is to treat a temporally structured sequence of latent actions as *future intentions*. Here, *intention* is an operational term for the predicted sequence of transition targets available to the action expert. This sequence encodes task-relevant transitions that should occur, without reconstructing all visual details of the future. LAWA jointly trains video, latent action, and action denoising experts through multi-model joint attention. A structured attention mask prevents access to future observations while allowing the action expert to use the evolving latent intention. At inference, LAWA discards the future-video branch and jointly denoises only latent intentions and action chunks. It therefore retains future-aware action generation without performing expensive visual prediction.

Learning a compact intention space introduces a second challenge: video reconstruction can favor static appearance over the small interaction dynamics that determine manipulation success. We address this issue with a discrete latent action tokenizer trained on action-free robot and egocentric videos. The tokenizer compresses visual transitions into discrete latent actions, while a forward decoder provides a next-observation reconstruction objective during training. An auxiliary mask-prediction objective is designed to bias the resulting tokens toward hands, manipulators, and their interaction regions. We automatically generate the mask targets with SAM 2 (Ravi et al. 2025), thereby enabling scal-

*These authors contributed equally.
†Corresponding author.
‡Project Leader.

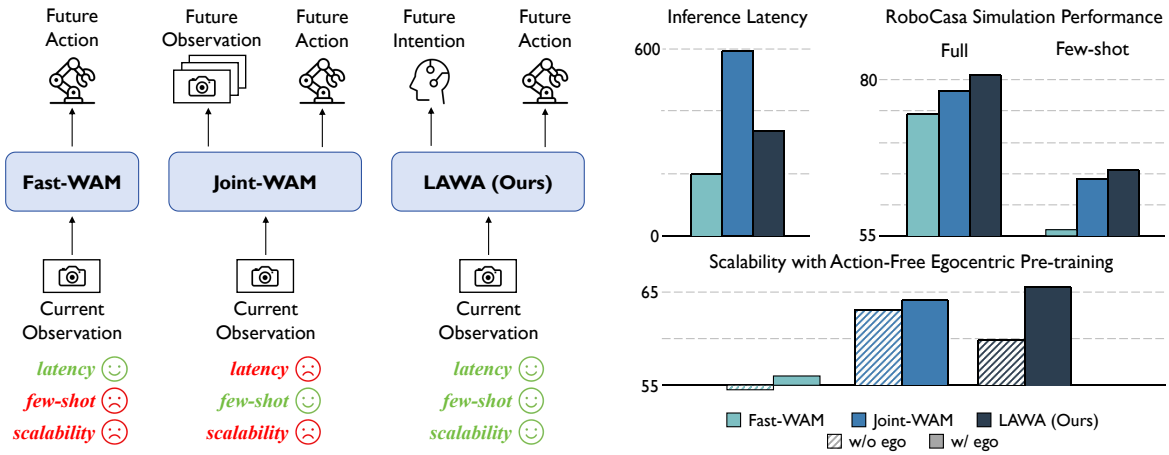


Figure 1: **Conceptual and quantitative comparison of WAM paradigms.** Fast-WAM, our primary performance baseline, removes test-time future imagination but generalizes worse; Joint-WAM predicts future observations and serves as the efficiency reference. LAWA treats latent actions as future intentions, substantially improving performance over Fast-WAM while matching Joint-WAM at lower latency. Moreover, LAWA obtains larger downstream gains from action-free data than both alternatives under matched video exposure and paradigm-specific pre-training objectives.

able annotation-free tokenizer training. We align motion-speed distributions across heterogeneous data sources and rebalance sampling to preserve robot-domain grounding. On RoboCasa, we empirically find that, without egocentric pre-training, the latent-action-based policy remains weaker than the matched Joint-WAM with explicit future-observation prediction, whereas scalable video pre-training reverses the gap. These results motivate compact latent actions as an efficient intention interface and show that action-free egocentric video provides complementary training diversity.

Experiments across few-shot, full-data, and zero-shot settings support this account. On RoboCasa, LAWA achieves state-of-the-art average success rates of 65.6% with only 10% of the training trajectories and 80.8% with full data, outperforming the matched Fast-WAM baseline by 9.6 and 4.5 points, respectively, while preserving the performance level of matched Joint-WAM. On LIBERO-Plus, LAWA reaches 74.4% zero-shot success under diverse perturbations, 14.4 points above the matched Fast-WAM and 4.0 points above Joint-WAM. On real-world tasks, LAWA demonstrates superior performance on both fine-grained assembly and long-horizon manipulation, surpassing full-data Fast-WAM with only 25% of the demonstrations. Despite retaining test-time future imagination, LAWA reduces inference latency by 42.9% relative to Joint-WAM while substantially outperforming the faster Fast-WAM. Controlled ablations further show larger downstream gains for LAWA from action-free egocentric pre-training than for both alternatives and continued improvement as the video corpus scales. This is a matched-video comparison: the paradigms use the same clips and preprocessing but their native objectives and trainable modules differ (see **Appendix**). Together, these results show that compact latent actions provide an effective balance among performance, generalization, and inference efficiency.

In summary, our contributions are as follows.

- We introduce LAWA, a WAM that treats compact latent actions as future intentions, retaining test-time future imagination without generating future observations.
- We develop a mask-supervised latent action tokenizer enhanced by scalable action-free pre-training.
- We demonstrate the effectiveness of our paradigm through extensive experiments across simulation benchmarks and real-world tasks, substantially improving performance over Fast-WAM while retaining Joint-WAM-level performance with lower inference latency.

## 2 Related Work

**World Action Models.** World action models (WAMs) augment visuomotor policies with predictive modeling so that action generation can account for how a scene may evolve (Zhu et al. 2025; Pai et al. 2025; Ye et al. 2026b; Kim et al. 2026b). Motus jointly models future images and actions (Bi et al. 2026), while DiT4DiT conditions action diffusion on video-denoising features (Ma et al. 2026). Generating future observations, however, incurs substantial inference cost. Fast-WAM therefore removes future prediction at test time while retaining video co-training (Yuan et al. 2026b), whereas Being-H0.7 aligns current-conditioned latent queries with a training-only future posterior (Luo et al. 2026). In contrast, LAWA retains *test-time future imagination* in a compact space: predicted latent actions serve as future intentions without future-observation generation.

**Latent Actions for Robotic Manipulation.** Latent actions compress visual transitions into action-like variables, enabling pre-training on videos without action labels (Schmidt and Jiang 2024; Liang et al. 2025a; Zhang et al. 2026b; Li et al. 2026b). LAPA quantizes inter-frame transitions into discrete codes for VLA pre-training and adapts them to robot actions with labeled demonstrations (Ye et al. 2025).

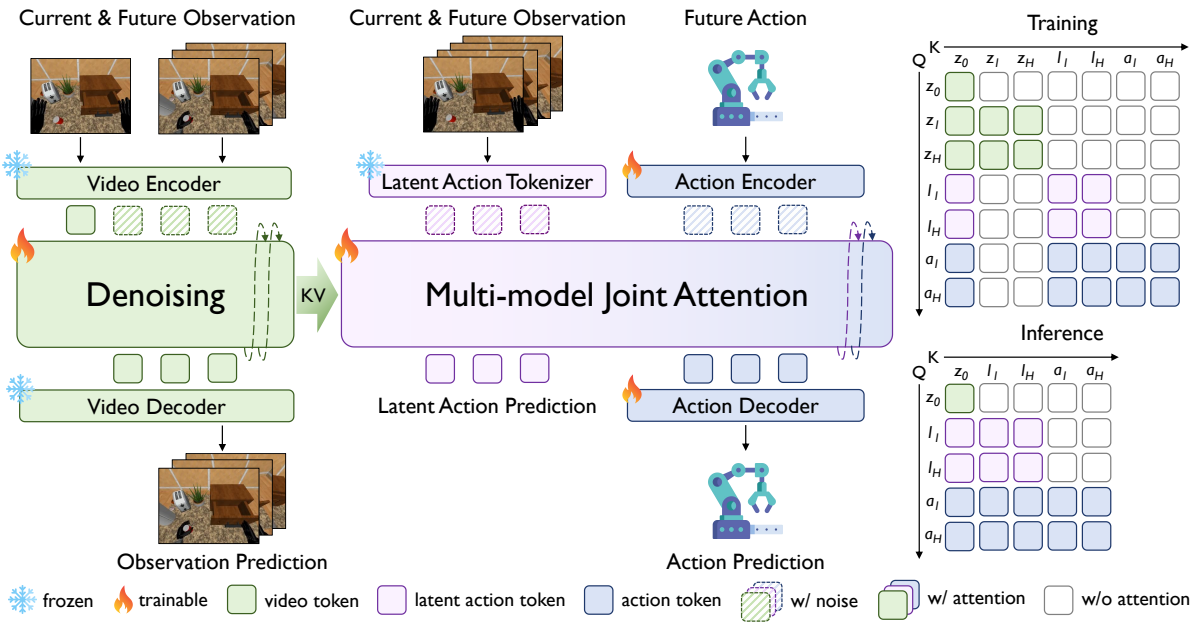


Figure 2: **Overview of our LAWA.** A structured mask controls information flow among the three branches. At inference, LAWA discards the future-video branch and jointly denoises compact latent intentions and action chunks, retaining future-aware action generation without explicitly predicting future observations. The language instruction is omitted for better illustration.

UniVLA learns task-centric latent actions (Bu et al. 2025); ViPRA grounds them via future-video prediction (Routray et al. 2026); and Motus leverages optical flow to learn motion-centric latent actions (Bi et al. 2026). DIAL, an end-to-end VLA in our taxonomy, predicts continuous visual foresight in a VLM space and uses a separate inverse-dynamics policy (Chen et al. 2026). Being-H0.7 transfers future information from a training-only posterior to deployable latent queries (Luo et al. 2026). LAWA instead predicts a ViPRA-style discrete transition sequence online and jointly denoises its continuous relaxation with executable actions at test time. **Appendix** provides a structured comparison.

## 3 Method

We consider a policy that maps the current observation $o_t$ and language instruction $c$ to an action chunk $a_{t+1:t+H}$. Fast-WAM (Yuan et al. 2026b) couples a pre-trained video Diffusion Transformer with an action expert through shared attention. During training, its branches denoise future-observation latents $z_{t+1:t+H}$ and predict continuous actions; a structured attention mask prevents the action tokens from accessing future-video tokens, so both branches learn from the common current observation tokens $z_t$ without future-information leakage. At inference, Fast-WAM encodes the current observation once and omits video denoising, gaining speed but losing an explicit representation of how the task should progress. In contrast to observation-level future imagination like Joint-WAM, our LAWA performs latent-action future imagination: as shown in Figure 2, it predicts a compact latent action sequence as future intentions and uses it to guide the action expert, thereby preserving an efficient inference interface like Fast-WAM.

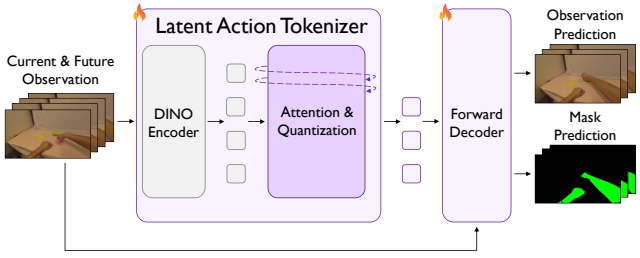


Figure 3: Action-free egocentric pre-training pipeline of the latent action tokenizer.

### 3.1 Latent Action Tokenizer

**Architecture.** Building upon ViPRA (Routray et al. 2026), the tokenizer maps an observation sequence $o_{t:t+H}$ to discrete latent action tokens $l_{t+1:t+H}$ that capture dynamic transitions without using action labels. A DINOv2 (Oquab et al. 2023) encoder first extracts patch tokens for every frame. A non-causal transformer then contextualizes these tokens through factorized spatial and temporal attention, producing $F_k \in \mathbb{R}^{N\times d}$ associated with each observation $o_k$. We encode the transition by differencing $F_k$ and $F_{k-1}$, and spatially compress the result into $L$ tokens $h_k \in \mathbb{R}^{L\times d_q}$. Each compressed token is assigned to its nearest entry in a learnable codebook $\mathcal{C} = \{e_j\}_{j=1}^{K}$. For $p = 1, \ldots, L$,

$$j^\star_{k,p} = \arg\min_j \|h_{k,p} - e_j\|_2^2,\ l_{k,p} = e_{j^\star_{k,p}}. \qquad (1)$$

To ground the codes in observable dynamics, a causal forward decoder concatenates the projected latent action $l_k$ as

additional spatial tokens with the patch tokens of $o_{k-1}$ and reconstructs $o_k$. Thus, the discrete representation must retain transition information that cannot be inferred from static appearance alone.

**Manipulation-centric Auxiliary Supervision.** Video reconstruction can be dominated by scene appearance, causing the tokenizer to underrepresent small but task-critical interactions. Motus (Bi et al. 2026) proposes optical flow as a suitable motion representation; however, we empirically find that mask prediction is more effective. We therefore augment the video-prediction tokenizer with a SAM-style mask decoder (Kirillov et al. 2023). For each transition, the decoder takes detached visual reconstruction tokens as image tokens and the projected latent action as a prompt, and predicts the corresponding hand or robot-manipulator mask. We automatically generate mask targets using SAM 2 (Ravi et al. 2025), thus avoiding laborious manual annotation and enabling scalable pre-training. This objective provides region-level supervision designed to bias the latent actions toward interaction regions. We evaluate its downstream effect in Table 6, without assuming that the auxiliary loss alone determines the semantics of the learned codes.

**Action-free Egocentric Pre-training.** Empirically, we observe that a tokenizer trained exclusively on robot data can capture transitions and provide compact latent action targets for future imagination, but the limited data diversity restricts its generalization capabilities. To address this issue, as shown in Figure 3, we leverage egocentric pre-training, which co-trains the tokenizer and forward decoder on action-free videos from both robot data and egocentric datasets. We apply per-source frame sampling during pre-training to align their motion speed distributions. This alignment reduces optimization interference caused by large differences in execution rates across heterogeneous sources, as also observed in Qwen-RobotManip (Yuan et al. 2026a). We further apply weighted rebalancing instead of sampling proportionally to dataset size, so that robot videos constitute approximately 20% of the expected training samples. Together, speed alignment and weighted sampling preserve robot-domain grounding while allowing the model to exploit diverse egocentric manipulation videos. See **Appendix** for data, optimization, and comparison details.

### 3.2 Latent Action as Intention

During robot-policy training, we freeze the pre-trained tokenizer and use it to encode $o_{t:t+H}$ into target latent actions $l_{t+1:t+H}$. LAWA jointly trains video, latent action, and action denoising experts, keeping the current observation clean while independently corrupting their future targets. At inference, LAWA omits future-video generation, encodes $o_t$ once, and jointly denoises the latent actions and executable actions from noise. The predicted latent states are trained toward codebook-derived transition targets and are used as future intentions for efficient future imagination. The tokenizer targets are discrete codebook embeddings, but the latent expert performs flow matching in their continuous embedding space. We do not apply nearest-neighbor projection during or after denoising; the action expert conditions on the evolving continuous latent state. Thus, *discrete* describes the tokenizer's target set, whereas test-time latent imagination is a continuous relaxation anchored to that set.

**Multi-model Joint Attention.** We couple the three experts through joint attention while retaining modality-specific Transformer blocks and timestep conditioning. As shown in Figure 2, through a structured attention mask, current-observation tokens $z_t$ cannot access the future, whereas noisy future-video tokens attend to all video tokens. Latent action tokens attend only to the current observation and latent action sequence; action tokens attend to the current observation, latent action sequence, and action sequence. This mask lets the action expert use the evolving future intention without leaking future-video information. All experts also condition on language and, when available, current proprioception. At inference, future-video tokens are omitted and the current-observation features are cached, so iterative denoising evaluates only the latent action and action experts. The same visibility mask is maintained across training and inference, while future-observation generation is avoided at test time.

### 3.3 Training Objective

Training proceeds in two stages. We first pre-train the latent action tokenizer on action-free videos using

$$\mathcal{L}_{\text{tok}} = \mathcal{L}_1 + \lambda_{\text{perc}}\mathcal{L}_{\text{perc}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}, \tag{2}$$

where $\mathcal{L}_1$ and the LPIPS-based $\mathcal{L}_{\text{perc}}$ supervise next-frame reconstruction (Zhang et al. 2018). For mask prediction, we utilize binary cross-entropy, Dice and IoU losses as $\mathcal{L}_{\text{mask}}$. During training, we adopt noise-substitution quantization (Vali and Bäckström 2022) to avoid additional codebook loss terms. Low-usage entries are periodically refreshed to prevent codebook collapse.

We then freeze the tokenizer and optimize the experts with flow matching. Let $y^m$ be the clean target for modality $m \in \{\text{vid}, \text{lat}, \text{act}\}$, corresponding to future-video latents, latent actions, and executable action chunks. For each branch, we independently sample Gaussian noise $\epsilon^m \sim \mathcal{N}(0, I)$ and a shifted flow time $\tau_m$, construct $y^m_{\tau_m} = (1-\tau_m)y^m + \tau_m\epsilon^m$, and set $v^m = \epsilon^m - y^m$. Let $\hat{v}^m_\theta$ denote the predicted velocity conditioned on $y^m_{\tau_m}$, $\tau_m$, and the context $\mathcal{C}_m$ permitted by multi-model joint attention. We minimize

$$\mathcal{L}_m = \mathbb{E}\Big[w_m(\tau_m)\,\|\hat{v}^m_\theta - v^m\|_2^2\Big], \tag{3}$$

where $w_m(\tau_m)$ is the scheduler-dependent training weight. We mask padded targets when computing their respective losses. The robot-policy training objective is

$$\mathcal{L}_{\text{LAWA}} = \lambda_{\text{vid}}\mathcal{L}_{\text{vid}} + \lambda_{\text{lat}}\mathcal{L}_{\text{lat}} + \lambda_{\text{act}}\mathcal{L}_{\text{act}}. \tag{4}$$

The video term preserves the world-modeling signal, the latent term matches codebook-derived transition targets, and the action term learns executable control.

## 4 Experiments

### 4.1 Experimental Setup

**RoboCasa.** Following the protocol of DIAL (Chen et al. 2026), we evaluate our method on RoboCasa (Nasiriany et al.

| Method | Paradigm | SR (%) Few-shot | Full |
|---|---|---|---|
| GR00T N1.6 (Bjorck et al.) | VLA | - | 47.6 |
| StarVLA (Community) | VLA | - | 48.8 |
| StarVLA-$\alpha$ (Ye et al.) | VLA | - | 53.8 |
| TwinBrainVLA (Yu et al.) | VLA | - | 54.6 |
| ABot-M0 (Yang et al.) | VLA | - | 58.3 |
| RLDX-1 (Kim et al.) | VLA | - | 58.7 |
| JoyAI-RA (Zhang et al.) | VLA | - | 63.2 |
| DIAL (Chen et al.) | VLA | 58.3 | 70.2 |
| Being-H0.7 (Luo et al.) | WAM | - | 49.2 |
| DiT4DiT (Ma et al.) | WAM | - | 50.8 |
| LDA-1B (Lyu et al.) | WAM | - | 55.4 |
| Fast-WAM† (Yuan et al.) | WAM | 56.0 | 76.3 |
| Joint-WAM† | WAM | 64.1 | 78.8 |
| **LAWA (Ours)** | WAM | **65.6** | **80.8** |

Table 1: **Results on RoboCasa benchmark.** Success rates (SR) across 24 tabletop tasks in both few-shot and full data settings. Best results in **bold**. † denotes our implementation.

2024), a simulation benchmark containing 24 tabletop rearrangement and articulated-object tasks. The *full* data setting uses 24,000 trajectories (1,000 per task) for training, whereas the *few-shot* setting utilizes a 10% subset. The latter therefore tests whether a model can benefit from test-time future imagination when robot supervision is scarce. We report the average success rates over 50 trials per task.

**LIBERO-Plus.** We further evaluate zero-shot transfer performance on LIBERO-Plus (Fei et al. 2026), which expands LIBERO (Liu et al. 2023) with challenging environmental perturbations. Following prior work, we report average success rate for each perturbation across all four task suites and their micro average. All models are trained solely on the original LIBERO training trajectories and receive no fine-tuning on the augmented LIBERO-Plus training data.

**Implementation Details.** Our Fast-WAM and Joint-WAM implementations and LAWA share downstream splits, optimization, batch sizes, and training steps, but not paradigm-specific branches, objectives, parameter counts, or inference costs. Fast-WAM is the primary performance baseline; Joint-WAM measures the cost of explicit future generation. We measure latency per action-chunk prediction on identical hardware and defer full details to the **Appendix**.

## 4.2 Performance on Simulation Benchmarks

**Main Results on RoboCasa.** As shown in Table 1, we compare LAWA against several state-of-the-art VLA and WAM methods. LAWA achieves the best results in both few-shot and full-data settings. It reaches 65.6% with 100 demonstrations per task and 80.8% with full data, exceeding matched Fast-WAM by 9.6 and 4.5 points and attaining comparable success to Joint-WAM (64.1% and 78.8%). The two comparisons establish the complete system's performance gain and its retention of observation-imagination

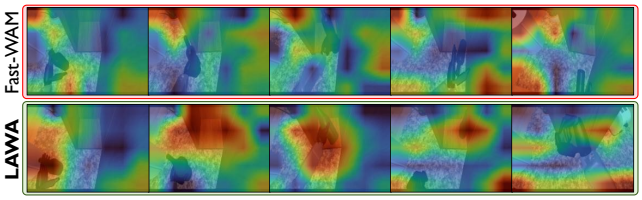


Figure 4: **Action-to-vision attention map comparison.** In this illustrative rollout, LAWA concentrates more on the manipulated object and interaction region, rather than diffuse background appearance.

performance, respectively. LAWA also exceeds DIAL by 7.3 points in few-shot, surpassing all reported WAMs and most full-data VLA baselines. Together with the studies below, these results support using a compact latent sequence as a future-intention interface without computationally expensive future-observation generation.

**Zero-shot Transfer Results on LIBERO-Plus.** As shown in Table 2, LAWA achieves a competitive overall success rate of 74.4% on LIBERO-Plus, surpassing OpenVLA-OFT by 4.8 points and matched Joint-WAM's 70.4%. Under the aligned training recipe, our matched Fast-WAM implementation reaches 60.0%, whereas LAWA reaches 74.4%, outperforming it by 14.4 points and achieving higher success across all six non-linguistic observation- and proprioceptive-level perturbations. The gains are especially pronounced under camera-viewpoint and sensor-noise shifts, reaching 44.3 and 27.5 points, respectively. This controlled system-level gap is consistent with the value of retaining a test-time future representation under distribution shift. LAWA realizes that representation with compact latent actions rather than explicit future observations.

## 4.3 Comparison between Paradigms

**The Effectiveness of Latent Action.** We first test whether the executor functionally uses the predicted latent sequence by perturbing only its model-visible latent-action state at inference. As shown in Table 3, Gaussian noise reduces full-data RoboCasa success from 80.8% to 52.2%, while temporal shuffling yields 56.4%. These interventions demonstrate that action generation depends on both the content and temporal organization of the latent sequence, rather than ignoring this pathway. Figure 4 provides complementary qualitative evidence on RoboCasa *WineToCabinetClose*. Fast-WAM distributes attention broadly over the countertop and background and ultimately fails, whereas LAWA tracks the manipulated object and task-relevant region across successive interaction stages and completes the task. Together, the intervention and visualization show that temporally structured latent actions provide a functionally used, manipulation-focused conditioning signal for action generation.

**The Effectiveness of Egocentric Pre-training.** Table 4 shows that action-free egocentric pre-training benefits all three paradigms, but to markedly different degrees. Fast-WAM improves by 1.5 and 1.7 points and Joint-WAM by

| Method | Paradigm | Perturbation Type | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|
| | | Camera | Robot | Language | Light | Background | Noise | Layout | |
| OpenVLA (Kim et al.) | VLA | 0.8 | 3.5 | 23.0 | 8.1 | 34.8 | 15.2 | 28.5 | 15.6 |
| OpenVLA-OFT (Kim, Finn, and Liang) | VLA | 56.4 | 31.9 | 79.5 | 88.7 | **93.3** | 75.8 | 74.2 | 69.6 |
| UniVLA (Bu et al.) | VLA | 1.8 | 46.2 | 69.6 | 69.0 | 81.0 | 21.2 | 31.9 | 42.9 |
| WorldVLA (Cen et al.) | VLA | 0.1 | 27.9 | 41.6 | 43.7 | 17.1 | 10.9 | 38.0 | 25.0 |
| $\pi_0$ (Black et al.) | VLA | 13.8 | 6.0 | 58.8 | 85.0 | 81.4 | 79.0 | 68.9 | 53.6 |
| $\pi_0$-FAST (Pertsch et al.) | VLA | 65.1 | 21.6 | 61.0 | 73.2 | 73.2 | 74.4 | 68.8 | 61.6 |
| Fast-WAM† (Yuan et al.) | WAM | 24.9 | 50.7 | 76.9 | 89.2 | 62.3 | 58.0 | 67.7 | 60.0 |
| Joint-WAM† | WAM | 47.2 | 65.3 | **91.8** | 94.8 | 57.6 | 61.0 | **78.9** | 70.4 |
| **LAWA (Ours)** | WAM | **69.2** | **66.7** | 62.8 | **96.2** | 64.5 | **85.5** | 78.6 | **74.4** |

Table 2: **Zero-shot transfer results on LIBERO-Plus benchmark.** Success rates under seven perturbations and their micro average. Best results in **bold**. † denotes our implementation.

| Input | Unperturbed | Gaussian | Temporal Shuffle |
|---|---|---|---|
| SR (%) | **80.8** | 52.2 | 56.4 |

Table 3: **Inference-time latent-action perturbation results.** Gaussian denotes isotropic noise with $\sigma = 1.0$.

| Method | Few-shot SR (%) | | Full SR (%) | |
|---|---|---|---|---|
| | w/o ego | w/ ego | w/o ego | w/ ego |
| Fast-WAM | 54.5 | 56.0 | 74.6 | 76.3 |
| Joint-WAM | **63.1** | 64.1 | **78.3** | 78.8 |
| **LAWA (Ours)** | 59.7 | **65.6** | 76.3 | **80.8** |

Table 4: Egocentric pre-training effectiveness comparison.

| Method | Fast-WAM | Joint-WAM | LAWA |
|---|---|---|---|
| Latency (ms) | 196.5 | 593.1 | 338.5 |

Table 5: **End-to-end inference latency comparison.** Results measured on one NVIDIA A800 GPU.

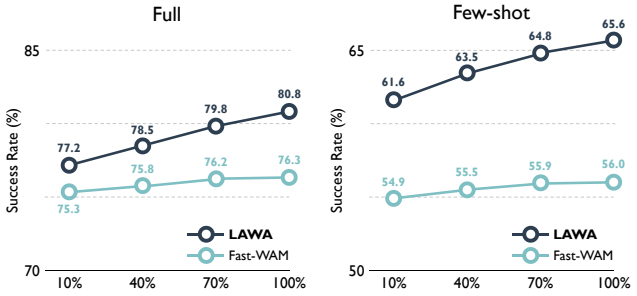


Figure 5: **Egocentric pre-training scalability comparison.** Under matched video fractions, LAWA obtains larger downstream gains than Fast-WAM in both data settings.

only 1.0 and 0.5 points in the few-shot and full data settings, whereas LAWA gains 5.9 and 4.5 points, respectively. Notably, without egocentric pre-training, LAWA does not yet match Joint-WAM: it trails by 3.4 and 2.0 points. Thus, compact latent-action prediction alone is not uniformly stronger than explicit future-observation prediction. After pre-training, however, LAWA reaches 65.6% and 80.8% on RoboCasa, matching the Joint-WAM performance reference while retaining its inference-efficiency advantage. Under this matched-video protocol, each paradigm uses its native objective and trainable modules. The results therefore establish larger downstream gains for LAWA under the stated schedules. A plausible explanation is that the compact bottleneck reduces appearance shortcuts while mask supervision emphasizes interaction regions (see Table 6).

**The Scalability of Egocentric Pre-training.** Figure 5 reveals a clear difference in scalability. Increasing the fraction of action-free egocentric videos from 10% to 100% raises LAWA from 77.2% to 80.8% in full-data and from 61.6% to 65.6% in few-shot setting, corresponding to gains of 3.6 and 4.0 points. Under the corresponding pre-training schedule, Fast-WAM improves by only 1.0 and 1.1 points. The widening downstream gap shows that LAWA continues to benefit as the video corpus grows, with the largest gain when labeled robot demonstrations are scarce.

**Performance–latency Trade-off.** As shown in Table 5, LAWA requires 338.5 ms per action chunk, 42.9% below Joint-WAM's 593.1 ms at comparable success. Fast-WAM is faster at 196.5 ms, but this advantage comes with substantially lower success rates. LAWA therefore improves performance and generalization over Fast-WAM, while preserving Joint-WAM-level performance at lower latency. This complementary comparison defines the favorable performance–latency trade-off of latent-action future imagination.

## 4.4 Ablation Study

We performed ablation studies of the proposed components in Table 6 by progressively adding latent actions (LA), egocentric pre-training (EP), and an auxiliary loss (AL). Introducing latent actions alone improves the Fast-WAM baseline by 5.2 and 1.7 percentage points in the few-shot and full data settings, respectively. Action-free egocentric pre-

| LA | EP | AL | SR (%) Few-shot | SR (%) Full |
|---|---|---|---|---|
| | | | 54.5 | 74.6 |
| ✓ | | | 59.7 | 76.3 |
| ✓ | ✓ | | 64.8 | 79.3 |
| ✓ | ✓ | Flow | 63.5 | 78.6 |
| ✓ | ✓ | Mask | **65.6** | **80.8** |

Table 6: Component ablation results, including Latent Action (LA), Egocentric Pre-training (EP) and Auxiliary Loss (AL).

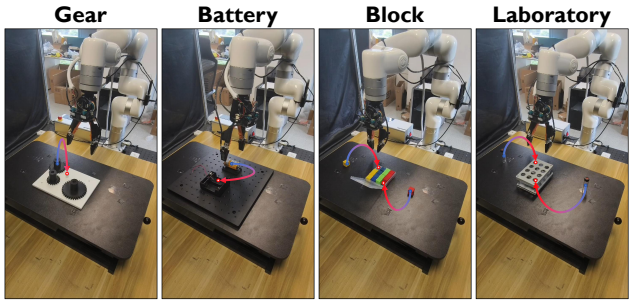


Figure 6: **Real-world tasks.** *Gear* and *Battery* involve assembly with fine-grained operations, while *Block* and *Laboratory* require long-horizon manipulation. Colored arrows indicate the required object transfers from their initial locations (blue) to their target locations (red).

training contributes a further 5.1 and 3.0 points. These gains show that the latent branch and egocentric pre-training provide complementary system-level improvements. Among the auxiliary targets, we empirically find mask prediction to be more effective than optical flow prediction. Relative to using no auxiliary loss, flow prediction degrades performance by 1.3 and 0.7 points in the two settings, whereas mask prediction improves performance by 0.8 and 1.5 points. Overall, the component additions are empirically complementary, and the mask-supervised tokenizer yields a further modest improvement. We interpret the mask objective as a manipulation-oriented inductive bias, rather than direct proof of the learned representation's semantics.

### 4.5 Performance on Real-World Tasks

**Setup and tasks.** We conduct real-world evaluation using a UFACTORY xArm7 equipped with a gripper. Besides, we use a RealSense D435 camera providing the base view and two fisheye cameras providing wrist views for visual inputs. We design four real-world tasks to cover complementary manipulation requirements, as illustrated in Figure 6. *Gear* and *Battery* are assembly tasks that require relatively fine-grained operations and are therefore more challenging. For example, the robot must adjust the gear orientation so that its teeth mesh with those of the adjacent gear, or press the battery until it is fully seated in the slot. In contrast, *Block* and *Laboratory* involve multi-stage manipulation over long horizons. For each task, we collect 200 demonstration trajectories for training and evaluate each policy over 20 trials.

| Method | Data | Gear | Batt. | Block | Lab. | Avg. |
|---|---|---|---|---|---|---|
| Fast-WAM | 25% | 30.0 | 5.0 | 0.0 | 0.0 | 8.8 |
| **LAWA (Ours)** | 25% | **55.0** | **30.0** | **45.0** | **30.0** | **40.0** |
| Fast-WAM | 50% | 50.0 | 20.0 | 10.0 | 0.0 | 20.0 |
| **LAWA (Ours)** | 50% | **70.0** | **50.0** | **60.0** | **45.0** | **56.3** |
| Fast-WAM | 100% | 65.0 | 40.0 | 25.0 | 5.0 | 33.8 |
| **LAWA (Ours)** | 100% | **85.0** | **65.0** | **70.0** | **50.0** | **67.5** |

Table 7: **Results on real-world tasks.** Success rates across each task under different proportions of robot demonstration trajectories. Best results of each data setting in **bold**.

**Results.** Table 7 shows that LAWA consistently outperforms Fast-WAM across all four tasks and all data settings. With 25%, 50%, and 100% of the demonstrations, LAWA improves over the Fast-WAM baseline by 31.2, 36.3, and 33.8 points, respectively. The largest gains in the full-data setting occur on the long-horizon tasks, where LAWA outperforms Fast-WAM by 45 points on both *Block* and *Laboratory*, consistent with the hypothesis that LAWA's test-time latent pathway benefits multi-stage execution. Moreover, LAWA also performs strongly on the fine-grained assembly tasks, surpassing Fast-WAM by 20 and 25 points on *Gear* and *Battery*, suggesting that the mask-supervised tokenizer may provide useful guidance for action generation.

LAWA remains effective when robot demonstrations are scarce. Using only 25% of the data, corresponding to 50 trajectories per task, LAWA attains an average success rate of 40.0%, surpassing the 33.8% achieved by Fast-WAM with the full training set. The advantage is particularly pronounced on long-horizon manipulation: Fast-WAM records no successful trial on either *Block* or *Laboratory* in this few-shot setting, whereas LAWA reaches 45.0% and 30.0%, respectively. Together with Table 4, these results show that the complete LAWA system benefits from diverse action-free egocentric videos when labeled robot demonstrations are limited. See **Appendix** for qualitative comparisons.

## 5 Conclusion

We presented LAWA, a WAM that treats compact latent actions operationally as future intentions and jointly denoises them with executable actions, preserving future-aware control without observation prediction. Automatic mask supervision and action-free robot and egocentric videos enhance its discrete tokenizer. Across RoboCasa, LIBERO-Plus, and real-world tasks, LAWA improves over Fast-WAM while retaining Joint-WAM-level performance with 42.9% lower latency. Unlike Joint-WAMs that explicitly generate future observations, this compact pathway preserves future-aware action generation without costly visual prediction. These results show that future imagination need not be discarded for efficiency, but can instead be represented through compact latent actions. We hope LAWA establishes a practical WAM paradigm for efficient future imagination and scalable learn-

ing from action-free videos.

## References


Bi, H.; Tan, H.; Xie, S.; Wang, Z.; Huang, S.; Liu, H.; Zhao, R.; Feng, Y.; Xiang, C.; Rong, Y.; et al. 2026. Motus: A unified latent action world model. In *CVPR*.

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

Black, K.; Brown, N.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; Groom, L.; Hausman, K.; Ichter, B.; et al. 2025. $\pi_0$: A Vision-Language-Action Flow Model for General Robot Control. In *RSS*.

Bu, Q.; Yang, Y.; Cai, J.; Gao, S.; Ren, G.; Yao, M.; Luo, P.; and Li, H. 2025. Univla: Learning to act anywhere with task-centric latent actions. In *RSS*.

Cen, J.; Yu, C.; Yuan, H.; Jiang, Y.; Huang, S.; Guo, J.; Li, X.; Song, Y.; Luo, H.; Wang, F.; et al. 2025. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*.

Chen, Y.; Ge, Y.; Zhou, H.; Ding, M.; Ge, Y.; and Liu, X. 2026. Dial: Decoupling intent and action via latent world modeling for end-to-end vla. *arXiv preprint arXiv:2603.29844*.

Community, S. 2026. StarVLA: A Lego-like Codebase for Vision-Language-Action Model Developing. *arXiv preprint arXiv:2604.05014*.

Fei, S.; Wang, S.; Shi, J.; Dai, Z.; Cai, J.; Qian, P.; Ji, L.; He, X.; Zhang, S.; Fei, Z.; Fu, J.; Gong, J.; and Qiu, X. 2026. LIBERO-Plus: A Progressive Robustness Benchmark for Visual-Language-Action Models. In *CVPR*.

Kim, D.; Jang, H.; Koo, M.; Jang, S.; Kim, T.; Kim, B.; Yoon, B.; Jang, C.; Choi, D.; Han, D.; et al. 2026a. Rldx-1 technical report. *arXiv preprint arXiv:2605.03269*.

Kim, M. J.; Finn, C.; and Liang, P. 2025. Fine-tuning vision-language-action models: Optimizing speed and success. In *RSS*.

Kim, M. J.; Gao, Y.; Lin, T.-Y.; Lin, Y.-C.; Ge, Y.; Lam, G.; Liang, P.; Song, S.; Liu, M.-Y.; Finn, C.; et al. 2026b. Cosmos policy: Fine-tuning video models for visuomotor control and planning. In *ICLR*.

Kim, M. J.; Pertsch, K.; Karamcheti, S.; Xiao, T.; Balakrishna, A.; Nair, S.; Rafailov, R.; Foster, E.; Lam, G.; Sanketi, P.; et al. 2024. Openvla: An open-source vision-language-action model. In *CoRL*.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. In *ICCV*.

Li, L.; Zhang, Q.; Luo, Y.; Yang, S.; Wang, R.; Han, F.; Yu, M.; Gao, Z.; Xue, N.; Zhu, X.; et al. 2026a. Causal World Modeling for Robot Control. *arXiv preprint arXiv:2601.21998*.

Li, S.; Gao, Y.; Sadigh, D.; and Song, S. 2025. Unified video action model. In *RSS*.

Li, Y.; Zhang, B.; Gu, C.; Ma, Z.; Zhang, J.; Deng, J.; Zhu, X.; and Zhang, L. 2026b. From Imagined Futures to Executable Actions: Mixture of Latent Actions for Robot Manipulation. In *ICML*.

Liang, A.; Czempin, P.; Hong, M.; Zhou, Y.; Biyik, E.; and Tu, S. 2025a. Clam: Continuous latent action models for robot learning from unlabeled demonstrations. *arXiv preprint arXiv:2505.04999*.

Liang, J.; Tokmakov, P.; Liu, R.; Sudhakar, S.; Shah, P.; Ambrus, R.; and Vondrick, C. 2025b. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*.

Liao, Y.; Zhou, P.; Huang, S.; Yang, D.; Chen, S.; Jiang, Y.; Hu, Y.; Cai, J.; Liu, S.; Luo, J.; et al. 2025. Genie envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*.

Liu, B.; Zhu, Y.; Gao, C.; Feng, Y.; Liu, Q.; Zhu, Y.; and Stone, P. 2023. Libero: Benchmarking knowledge transfer for lifelong robot learning. In *NeurIPS*.

Luo, H.; Zhang, W.; Feng, Y.; Zheng, S.; Xu, H.; Xu, C.; Xi, Z.; Fu, Y.; and Lu, Z. 2026. Being-h0.7: A latent world-action model from egocentric videos. *arXiv preprint arXiv:2605.00078*.

Lyu, J.; Liu, K.; Zhang, X.; Liao, H.; Feng, Y.; Zhu, W.; Shen, T.; Chen, J.; Zhang, J.; Dong, Y.; et al. 2026. Lda-1b: Scaling latent dynamics action model via universal embodied data ingestion. In *RSS*.

Ma, T.; Zheng, J.; Wang, Z.; Jiang, C.; Cui, A.; Liang, J.; and Yang, S. 2026. Dit4dit: Jointly modeling video dynamics and actions for generalizable robot control. *arXiv preprint arXiv:2603.10448*.

Nasiriany, S.; Maddukuri, A.; Zhang, L.; Parikh, A.; Lo, A.; Joshi, A.; Mandlekar, A.; and Zhu, Y. 2024. RoboCasa: Large-Scale Simulation of Everyday Tasks for Generalist Robots. In *RSS*.

Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Pai, J.; Achenbach, L.; Montesinos, V.; Forrai, B.; Mees, O.; and Nava, E. 2025. mimic-video: Video-action models for generalizable robot control beyond vlas. *arXiv preprint arXiv:2512.15692*.

Pertsch, K.; Stachowicz, K.; Ichter, B.; Driess, D.; Nair, S.; Vuong, Q.; Mees, O.; Finn, C.; and Levine, S. 2025. Fast: Efficient action tokenization for vision-language-action models. In *RSS*.

Ravi, N.; Gabeur, V.; Hu, Y.-T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; Mintun, E.; Pan, J.; Alwala, K. V.; Carion, N.; Wu, C.-Y.; Girshick, R.; Dollár, P.; and Feichtenhofer, C. 2025. SAM 2: Segment Anything in Images and Videos. In *ICLR*.

Routray, S.; Pan, H.; Jain, U.; Bahl, S.; and Pathak, D. 2026. ViPRA: Video Prediction for Robot Actions. In *ICLR*.

Schmidt, D.; and Jiang, M. 2024. Learning to act without actions. In *ICLR*.

Vali, M. H.; and Bäckström, T. 2022. NSVQ: Noise substitution in vector quantization for machine learning. *IEEE Access*.

Yang, Y.; Zeng, S.; Lin, T.; Chang, X.; Qi, D.; Xiao, J.; Liu, H.; Chen, R.; Chen, Y.; Huo, D.; et al. 2026. Abot-m0: Vla foundation model for robotic manipulation with action manifold learning. *arXiv preprint arXiv:2602.11236*.

Ye, J.; Gao, N.; Yang, S.; Zheng, J.; Wang, Z.; Chen, Y.; Chen, P.; Chen, Y.; Liu, S.; and Jia, J. 2026a. StarVLA-$\alpha$: Reducing Complexity in Vision-Language-Action Systems. In *ECCV*.

Ye, S.; Ge, Y.; Zheng, K.; Gao, S.; Yu, S.; Kurian, G.; Indupuru, S.; Tan, Y. L.; Zhu, C.; Xiang, J.; et al. 2026b. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*.

Ye, S.; Jang, J.; Jeon, B.; Joo, S. J.; Yang, J.; Peng, B.; Mandlekar, A.; Tan, R.; Chao, Y.-W.; Lin, B. Y.; et al. 2025. Latent action pretraining from videos. In *ICLR*.

Yu, B.; Lian, S.; Lin, X.; Wei, Y.; Shen, Z.; Wu, C.; Miao, Y.; Wang, X.; Wang, B.; Huang, C.; et al. 2026a. TwinBrainVLA: Unleashing the Potential of Generalist VLMs for Embodied Tasks via Asymmetric Mixture-of-Transformers. *arXiv preprint arXiv:2601.14133*.

Yu, H.; Lin, H.; Zhang, J.; Zhang, W.; Gu, C.; Li, H.; and Tan, P. 2026b. Maskwam: Unifying mask prompting and prediction for world-action models. *arXiv preprint arXiv:2606.13515*.

Yuan, H.; Liang, Z.; Chen, A.; Wang, Y.; Li, H.; Lin, P.; Huang, Y.; Lei, Z.; Zhang, T.; Zhang, J.; et al. 2026a. Qwen-robotmanip technical report: Alignment unlocks scale for robotic manipulation foundation models. *arXiv preprint arXiv:2606.17846*.

Yuan, T.; Dong, Z.; Liu, Y.; and Zhao, H. 2026b. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*.

Zhang, T.; Yuan, Z.; Chi, D.; Liu, P.; Li, D.; Hu, K.; Zhang, L.; Nie, J.; Wei, Z.; Chen, Z.; et al. 2026a. Joyai-ra 0.1: A foundation model for robotic autonomy. *arXiv preprint arXiv:2604.20100*.

Zhang, W.; Zhang, B.; Qi, Z.; Zeng, W.; Jin, X.; and Zhang, L. 2026b. Disentangled robot learning via separate forward and inverse dynamics pretraining. In *ICLR*.

Zhu, C.; Yu, R.; Feng, S.; Burchfiel, B.; Shah, P.; and Gupta, A. 2025. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *RSS*.