SPATIALWAM: SPATIALLY GROUNDED WORLD ACTION MODELS VIA EXPLICIT 3D GEOMETRIC LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

World action models inherit spatio-temporal priors from video models, but visual plausibility does not ensure geometric accuracy for fine-grained manipulation. We present **SpatialWAM**, which learns spatially grounded video and spatial representations through explicit 3D geometric learning for action generation. During video–spatial pre-training, a spatial expert extracts features from current and future video tokens using spatial and dynamics queries. Spatio-Temporal Gaussian Modeling (STGM) then fuses these features to predict current and future 3D Gaussian scenes shared across views. Multi-view depth supervision in turn shapes both experts' representations through bidirectional attention, using calibrated RGB-D videos without action annotations. During post-training, a randomly initialized action expert accesses the learned representations through joint attention, while all three experts are fully updated with continued video and depth supervision. At inference, all three experts remain active, while STGM is omitted. SpatialWAM achieves 73.4% success on RoboCasa, 97.8% on LIBERO, and 85.9% on LIBERO-Plus, with 72.0% average success across three real-world tasks. Comparisons using identical pre-training data show gains over video-only and 2D depth pre-training, supporting geometric learning and shared 3D scene modeling.

1 INTRODUCTION

Robotic manipulation increasingly benefits from the visual and semantic knowledge in pre-trained foundation models. Vision–Language–Action (VLA) models, such as OpenVLA-OFT (Kim et al., 2025a) and $\pi_{0.5}$ (Black et al., 2025a), transfer visual and semantic priors from pre-trained vision–language backbones to robotic manipulation. To capture how scenes evolve during interaction, world action models (WAMs), including Cosmos Policy (Kim et al., 2026), LingBot-VA (Li et al., 2026c), and DreamZero (Ye et al., 2026), couple future visual prediction with action generation using pre-trained video models. These models inherit rich spatio-temporal priors, but visually plausible predictions do not necessarily preserve the geometric accuracy needed for fine-grained manipulation. For example, inserting a grasped straw into a cup requires accurate estimation of their relative position and orientation to guide alignment (Figure 1). Such interactions motivate learning representations that capture both scene dynamics and the spatial structure relevant to action generation.

To improve spatial reasoning in such interactions, geometry-aware VLAs incorporate 3D cues (Qu et al., 2025), align features with geometric models (Li et al., 2026b), or predict future geometry (Qian et al., 2026). These vision–language backbones are not pre-trained for future visual prediction and lack video-derived dynamics priors for modeling scene evolution. Recent geometry-aware WAMs incorporate these priors into geometric learning: X-WAM (Guo et al., 2026) and WAM4D (Li et al., 2026d) predict future depth, while WSA_1 (Jiang et al., 2026) predicts geometric latents. However, these objectives do not explicitly couple geometric predictions across views through shared 3D scenes, allowing cross-view inconsistencies that can compromise precise action generation.

To address this gap, we propose **SpatialWAM** (Figure 1), a world action model that learns spatially grounded video and spatial representations through explicit 3D scene modeling. To this end, we introduce a spatial expert that attends to current and future video tokens during denoising, drawing on the spatio-temporal priors of a pre-trained video model to capture scene structure and its evolution. To impose a common geometric constraint across views, we decode the expert's output features into

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

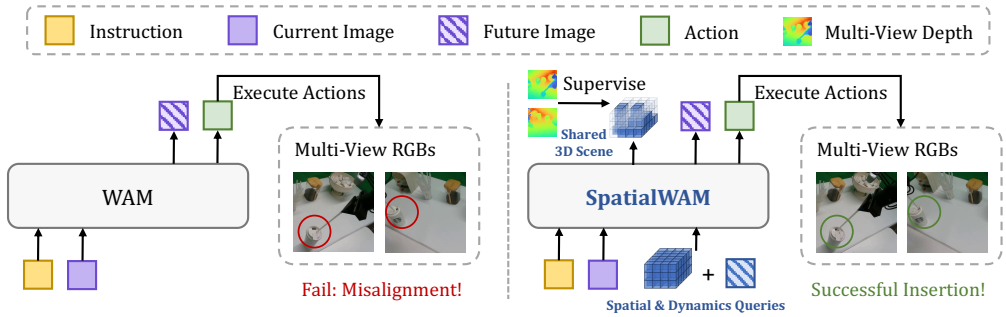

Figure 1: **Motivation and Overview of SpatialWAM.** During training, multi-view depth supervision of shared 3D scenes shapes video and spatial representations for action generation. Real-world straw-insertion examples show misalignment with Cosmos Policy (Kim et al., 2026) (left) and successful insertion with SpatialWAM (right), illustrating the need for accurate spatial alignment.

current and future 3D Gaussian scenes, each rendered from multiple viewpoints for depth supervision. This supervision, in turn, shapes both video and spatial representations through bidirectional interaction between the two experts, and the resulting features jointly guide action generation.

Specifically, SpatialWAM uses two-stage training: video–spatial pre-training for representation learning, followed by post-training for action generation. During pre-training, the spatial expert uses spatial and dynamics queries to extract scene structure and temporal dynamics from video tokens through bidirectional joint attention with the video expert. The Spatio-Temporal Gaussian Modeling (STGM) module then fuses these features to predict current and future 3D Gaussian scenes, each shared across views so that multi-view depth supervision constrains a common underlying geometry. We jointly optimize both experts and STGM using video prediction and depth supervision losses, allowing geometric learning to shape their representations before action post-training. Because this stage requires no action annotations, it can use calibrated multi-view RGB-D videos, offering a path to scaling representation learning beyond action-labeled robot demonstrations.

To translate these representations into actions, we introduce a randomly initialized action expert during post-training. Through tri-expert joint attention, action tokens attend to features from the video and spatial experts. We jointly optimize all three experts on robot demonstrations, adding an action prediction loss while retaining the video prediction and depth supervision losses. At inference, all three experts remain active for joint video–action denoising, while STGM is omitted, allowing action generation to use the learned spatial features without decoding Gaussian scenes.

SpatialWAM achieves 73.4%, 97.8%, and 85.9% success on RoboCasa (Nasiriany et al., 2024), LIBERO (Liu et al., 2023), and LIBERO-Plus (Fei et al., 2026), respectively. Across three real-world tasks, it achieves 72.0% average success, exceeding Cosmos Policy (Kim et al., 2026) by 50.0 percentage points. With identical pre-training data, SpatialWAM outperforms *Video-Only Pre-Training* and *2D Depth Pre-Training* on RoboCasa by 5.3 and 4.1 percentage points, respectively. Further data scaling and qualitative analyses suggest improved post-training data efficiency and video predictions that better reflect scene structure and gripper–object interactions.

In summary, our contributions are threefold: (1) We introduce **SpatialWAM**, a world action model that combines video dynamics priors with explicit 3D geometric learning and enables action tokens to access video and spatial representations through joint attention. (2) We design a spatial expert whose spatial and dynamics queries extract scene structure and dynamics from video tokens. STGM fuses these features to predict current and future 3D Gaussian scenes shared across views, providing a common geometric constraint through multi-view depth supervision and enabling pre-training without action annotations. (3) We demonstrate strong simulation and real-world manipulation performance, with comparisons using identical pre-training data supporting geometric learning and shared 3D scene modeling, and analyses suggesting improved post-training data efficiency.

2 Related Work

2.1 Vision–Language–Action and World Action Models

Pre-trained foundation models increasingly serve as the backbone of robotic manipulation policies. Vision–language–action (VLA) models, including RT-2 (Zitkovich et al., 2023), OpenVLA (Kim et al., 2025b), π_0 (Black et al., 2025b), UniVLA (Bu et al., 2025), and AVA-VLA (Xiao et al., 2026),

adapt pre-trained vision–language backbones using robot demonstrations for language-conditioned visuomotor control. Their backbones are pre-trained primarily on static image–text pairs, emphasizing semantic knowledge rather than the spatio-temporal dynamics learned through video prediction. SpatialWAM builds on a pre-trained video model to bring these dynamics priors to action generation.

World action models (WAMs) couple action learning with future visual prediction using video generation backbones. VideoVLA (Shen et al., 2025) jointly predicts actions and visual outcomes, while Motus (Bi et al., 2026) uses a Mixture-of-Transformers architecture for video–action generation. LingBot-VA (Li et al., 2026c) and DreamZero (Ye et al., 2026) support closed-loop control through autoregressive video–action modeling. Fast-WAM (Yuan et al., 2026) retains video prediction during training but omits future video generation at inference. However, RGB video prediction alone provides limited spatial reasoning, which is important for robotic manipulation. SpatialWAM learns spatially informed representations for action generation through explicit 3D geometric learning.

2.2 Geometry-Aware Robot Learning

Geometry-aware VLAs incorporate spatial information by enriching visual inputs or introducing geometric supervision. SpatialVLA (Qu et al., 2025) and PointVLA (Li et al., 2026a) enrich policy inputs with geometry derived from depth maps or point clouds. Spatial Forcing (Li et al., 2026b) supervises intermediate VLA features through alignment with a pre-trained geometric model. DreamVLA (Zhang et al., 2025) and GeoPredict (Qian et al., 2026) use auxiliary prediction of future geometry to guide action learning. In GeoPredict, image tokens represent current observations, so future geometry must be inferred without access to future visual tokens. SpatialWAM's spatial and dynamics queries instead attend to both current and future video tokens during joint denoising, allowing geometric prediction to use evolving features of future frames. This couples explicit geometric learning with the spatio-temporal dynamics priors of a pre-trained video model.

Geometry-aware WAMs incorporate future geometry prediction into video–action learning. X-WAM (Guo et al., 2026) and WSA_1 (Jiang et al., 2026) use dedicated branches to predict future geometry, while WAM4D (Li et al., 2026d) decodes future depth from spatial register tokens using a pre-trained geometric head. Their geometric supervision targets depth maps or latent features without explicit multi-view rendering from a shared 3D scene. SpatialWAM renders each predicted current or future 3D Gaussian scene across views for depth supervision, enforcing a common geometric constraint. Bidirectional video–spatial attention lets geometric learning shape video representations, and action tokens attend to both experts' representations during post-training and inference.

3 Methodology

We propose **SpatialWAM**, a world action model that learns spatially grounded representations for action generation through explicit 3D geometric learning. As illustrated in Figure 2, we jointly train video and spatial experts during pre-training, with the Spatio-Temporal Gaussian Modeling (STGM) module providing explicit geometric supervision. During post-training, we introduce an action expert that leverages the spatially grounded video and spatial representations to improve action generation for robot control, and jointly optimize all three experts. We first define the task and video backbone (Sec. 3.1), then present STGM (Sec. 3.2), its integration into video–spatial pre-training (Sec. 3.3), and action post-training and inference (Sec. 3.4).

3.1 Preliminary

Problem Definition. Given a language instruction $\mathbf{L}$, current multi-view RGB observations $\mathbf{I}_t$, and the robot's proprioceptive state $\mathbf{S}_t$, a world action model (Ye et al., 2026; Li et al., 2026c) predicts future observations and an action chunk $\mathbf{A}_t = (\mathbf{a}_t, \dots, \mathbf{a}_{t+H-1})$, where H is the action chunk length. We use k to denote the temporal offset from the current time t. A temporal downsampling factor r specifies the interval between future prediction times, with offsets $\mathcal{K} = \{r, 2r, \dots, H\}$, where H is divisible by r. Future video frames $\mathbf{I}_{t+k}$ correspond to $k \in \mathcal{K}$. Geometric supervision additionally includes the current time $k = 0$. During training, depth maps $\mathbf{D}^c_{t+k}$, camera intrinsics $\mathbf{K}_c$, and extrinsics $\mathbf{E}_c$ supervise scene geometry from view c for $k \in \{0\} \cup \mathcal{K}$.

Video Generation Backbone. We initialize the video expert from Wan2.2-5B (Wan et al., 2025) and freeze its video VAE. The VAE encodes observed and future frames into a latent sequence $\mathbf{z}$, tokenized and processed by the video transformer with T5 language conditioning. Following flow matching (Lipman et al., 2023), we perturb the video latents along a linear path:

$$\mathbf{z}^{\tau_v} = (1 - \tau_v)\mathbf{z} + \tau_v \boldsymbol{\epsilon}_v, \qquad \mathbf{v}^*_v = \boldsymbol{\epsilon}_v - \mathbf{z}, \tag{1}$$

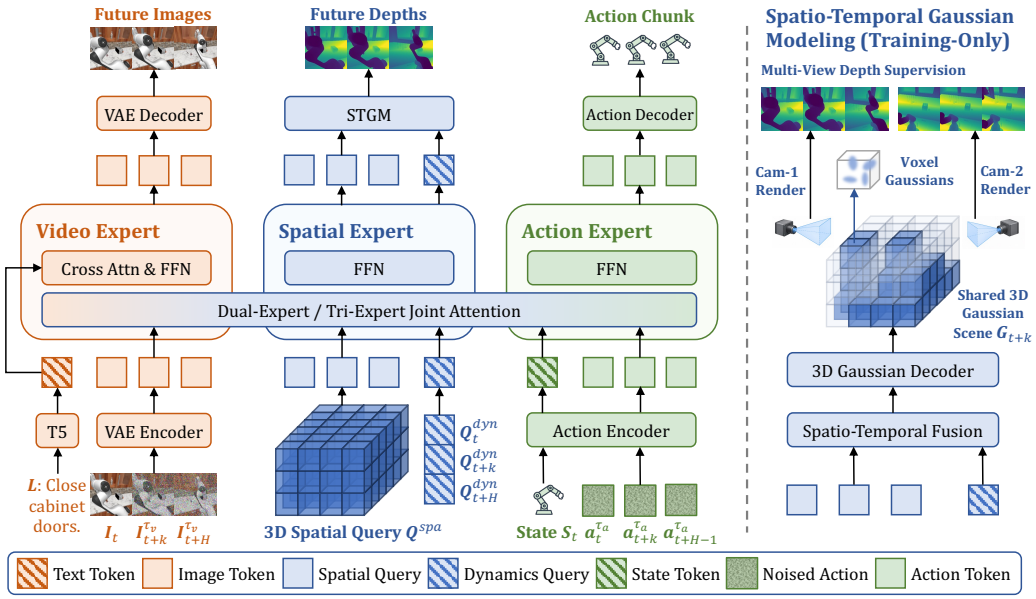

Figure 2: **Overview of SpatialWAM.** The Spatio-Temporal Gaussian Modeling (STGM) module (right) fuses spatial and dynamics features to predict current and future 3D Gaussian scenes, each shared across views for depth supervision. Dual-expert joint attention couples the video and spatial experts to incorporate this geometric learning into video–spatial pre-training. Post-training adds an action expert that accesses both video and spatial features through tri-expert joint attention. All three experts remain active during inference, while STGM is omitted.

where $\boldsymbol{\epsilon}_v \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\tau_v \in [0, 1]$ is the noise level, and $\mathbf{v}^*_v$ is the target velocity. Future latents share one sampled τ_v per sequence. Figure 2 depicts this latent-space corruption as noisy images. We replace the first latent frame with the clean encoding of $\mathbf{I}_t$, excluding its loss. Both stages minimize $\mathcal{L}_{\text{video}}$, the mean squared error between predicted and target velocities.

3.2 Spatio-Temporal Gaussian Modeling

The Spatio-Temporal Gaussian Modeling (STGM) module predicts current and future workspace geometry as 3D Gaussians to provide geometric priors for action generation. Geometric supervision propagates through bidirectional interaction between the spatial and video experts, shaping video representations to encode scene structure and its evolution. These representations promote geometrically consistent future image prediction and enhance spatial reasoning for precise manipulation. STGM uses spatio-temporal fusion, a 3D Gaussian Decoder, and multi-view depth supervision.

Spatial and Dynamics Queries. The spatial expert extracts features for STGM using spatial and dynamics queries. Spatial queries $\mathbf{Q}^{\text{spa}}$ represent the workspace on a coarse regular 3D grid with N_s query locations. Each location $\mathbf{p}_i$ is assigned a learnable embedding $\mathbf{e}_i$ and a 3D sinusoidal positional encoding. To model current and future scene geometry, we instantiate a dynamics query $\mathbf{Q}^{\text{dyn}}_{t+k}$ for each $k \in \{0\} \cup \mathcal{K}$ from a shared learnable embedding $\mathbf{e}_{\text{dyn}}$:

$$\mathbf{Q}^{\text{spa}}_i = \mathbf{e}_i + \mathcal{P}_{\text{3D}}(\mathbf{p}_i), \qquad \mathbf{Q}^{\text{dyn}}_{t+k} = \mathbf{e}_{\text{dyn}} + \mathcal{P}_{\text{1D}}(k). \tag{2}$$

Here, $\mathcal{P}_{\text{3D}}$ and $\mathcal{P}_{\text{1D}}$ denote sinusoidal spatial and temporal encoding, respectively. We collect the spatial queries into $\mathbf{Q}^{\text{spa}} = [\mathbf{Q}^{\text{spa}}_1; \ldots; \mathbf{Q}^{\text{spa}}_{N_s}]$. The spatial expert jointly processes both query types through bidirectional expert interaction (Sec. 3.3). The queries are not corrupted by diffusion noise, but their features depend on the video tokens through joint attention. We denote the resulting spatial and dynamics features by $\mathbf{h}_i$ and $\mathbf{h}^{\text{dyn}}_{t+k}$, respectively.

Spatio-Temporal Fusion and 3D Gaussian Decoder. For each offset $k \in \{0\} \cup \mathcal{K}$, we broadcast its dynamics feature to all workspace locations and fuse it with the local spatial feature:

$$\mathbf{f}_{t+k,i} = \mathbf{h}_i + \mathcal{F}_\phi\left([\mathbf{h}_i; \mathbf{h}^{\text{dyn}}_{t+k}]\right), \qquad \mathcal{G}_{t+k} = \mathcal{D}_{\text{GS}}\left(\{\mathbf{f}_{t+k,i}\}_{i=1}^{N_s}\right). \tag{3}$$

Here, $[\cdot;\cdot]$ denotes feature concatenation, $\mathcal{F}_{\boldsymbol{\phi}}$ is a two-layer MLP preceded by layer normalization with learnable parameters $\boldsymbol{\phi}$, and $\mathcal{D}_{\mathrm{GS}}$ is a shared 3D Gaussian Decoder. The decoder reshapes the fused features into a volume and uses transposed convolutions and upsampling to obtain a denser voxel grid. Each voxel predicts Gaussian primitives whose centers are anchored to the voxel center through bounded offsets. Their geometric attributes comprise position, opacity, rotation, and scale, following the 3D Gaussian representation of Kerbl et al. (2023). The union of these primitives forms $\boldsymbol{\mathcal{G}}_{t+k}$, the scene at time $t+k$. For each target time, we fuse its dynamics feature with the shared spatial features and use the same 3D Gaussian Decoder to predict the corresponding Gaussian scene.

Multi-View Depth Supervision. To encourage geometric consistency across views, we supervise each 3D Gaussian scene with multi-view depth. At time $t+k$, we render depth maps for all supervised views from the same scene using the corresponding camera parameters:

$$\widehat{\mathbf{D}}^{c}_{t+k} = \mathcal{R}_{\mathrm{depth}}(\boldsymbol{\mathcal{G}}_{t+k}; \mathbf{K}_c, \mathbf{E}_c)\,. \tag{4}$$

We restrict supervision to the robot workspace using a mask $\mathbf{M}^{c}_{t+k}$ that selects pixels whose back-projected 3D points lie within the workspace. The geometric objective is

$$\mathcal{L}_{\mathrm{depth}} = \operatorname*{mean}_{k\in\{0\}\cup\mathcal{K}} \operatorname*{mean}_{c} \frac{\sum_p \mathbf{M}^{c}_{t+k}(p)\, \mathcal{L}_{\mathrm{SL1}}\Big(\widehat{\mathbf{D}}^{c}_{t+k}(p) - \mathbf{D}^{c}_{t+k}(p)\Big)}{\sum_p \mathbf{M}^{c}_{t+k}(p)}, \tag{5}$$

where $\mathcal{L}_{\mathrm{SL1}}$ is the smooth L1 loss. Here, $\mathbf{M}^{c}_{t+k}(p)$, $\mathbf{D}^{c}_{t+k}(p)$, and $\widehat{\mathbf{D}}^{c}_{t+k}(p)$ are the corresponding values at pixel p. Supervising all $k \in \{0\}\cup\mathcal{K}$ trains the spatial representation to capture current scene structure and its evolution. Gradients propagate through rendering and the 3D Gaussian Decoder into the spatial expert and reach the video and action experts through joint attention.

3.3 Video–Spatial Pre-Training

We incorporate explicit 3D learning into video–spatial pre-training to shape video representations through geometric supervision before action learning. A randomly initialized spatial expert processes spatial and dynamics queries and interacts with the video expert through joint attention. STGM maps its output features to Gaussian scenes for the geometric objective in Eq. 5.

Bidirectional Expert Interaction. The video and spatial experts interact at every transformer layer through dual-expert joint attention. Each expert uses its own projections to map its tokens into a common query, key, and value space. For expert $e \in \mathcal{E}$, the attention output is

$$\mathbf{O}_e = \mathcal{A}(\mathbf{Q}_e, [\mathbf{K}_{e'}]_{e'\in\mathcal{E}}, [\mathbf{V}_{e'}]_{e'\in\mathcal{E}})\,, \qquad \mathcal{E} = \{\mathrm{video}, \mathrm{spatial}\}, \tag{6}$$

where $\mathcal{A}$ denotes attention and brackets $[\cdot]$ denote concatenation along the token dimension. Each expert projects its outputs back to its feature space and retains its own FFN. The video expert also cross-attends to language embeddings. Each dynamics query corresponds to a target frame time, temporally aligning geometric learning with video prediction. Through bidirectional interaction, spatial and dynamics queries access visual context, while video tokens incorporate geometric features encoding scene structure and its evolution.

Joint Pre-Training Objective. We jointly optimize video prediction and geometric learning:

$$\mathcal{L}_{\mathrm{pre}} = \lambda_{\mathrm{video}}\mathcal{L}_{\mathrm{video}} + \lambda_{\mathrm{depth}}\mathcal{L}_{\mathrm{depth}}. \tag{7}$$

The video term supervises visual dynamics, while the geometric term constrains current and future scenes through their shared 3D representations across views. We fully update both experts and jointly train STGM, while keeping the VAE fixed. This stage requires no action-prediction objective.

Pre-Training Data. We use approximately 252K single-arm trajectories totaling 850 hours from MimicGen (Mandlekar et al., 2023), RoboCasa365 (Nasiriany et al., 2026), and DROID (Khazatsky et al., 2024). MimicGen and RoboCasa365 provide simulation data, with multi-view depth, camera intrinsics and extrinsics, and language descriptions obtained directly from the simulator. For DROID, we use the geometric annotations provided in the PointWorld release (Huang et al., 2026). Further pre-training data details are provided in Appendix A.

3.4 Action Post-Training and Inference

Spatially Grounded Action Generation. We initialize the video and spatial experts and STGM from the pre-training checkpoint and introduce a randomly initialized action expert. Separate encoders map the current state and noisy actions into one state token and H action tokens with sequence positional encodings. We extend the expert set in Eq. 6 to $\mathcal{E} = \{\text{video}, \text{spatial}, \text{action}\}$, yielding tri-expert joint attention over all tokens. Action tokens attend to both spatially grounded video representations and spatial expert features throughout action denoising.

Joint Post-Training Objective. We perturb the demonstrated action chunk as $\mathbf{A}_t^{\tau_a} = (1-\tau_a)\mathbf{A}_t + \tau_a \boldsymbol{\epsilon}_a$, where $\boldsymbol{\epsilon}_a$ is Gaussian noise. For each sample, one action noise level τ_a is shared across the entire chunk and sampled independently of the video noise level τ_v. The action expert predicts the velocity $\boldsymbol{\epsilon}_a - \mathbf{A}_t$, with an MSE objective $\mathcal{L}_{\text{action}}$ evaluated on the action tokens, excluding the state token. We retain visual and geometric supervision during policy adaptation:

$$\mathcal{L}_{\text{post}} = \lambda_{\text{video}}\mathcal{L}_{\text{video}} + \lambda_{\text{depth}}\mathcal{L}_{\text{depth}} + \lambda_{\text{action}}\mathcal{L}_{\text{action}}. \tag{8}$$

All three experts and their prediction heads are fully updated on training data from the post-training benchmarks, while the VAE remains fixed. This objective transfers the learned representations to action prediction while preserving their visual and geometric training objectives.

Joint Inference. During inference, we initialize future video latents and the action chunk from Gaussian noise, keeping the current-frame latent fixed. All three experts jointly denoise video and actions using Euler updates along a shared noise-level schedule from one to zero. At each step, the spatial expert initializes spatial and dynamics queries from learned embeddings and positional encodings, with its output features entering joint attention. STGM is omitted. The model outputs an H-step action chunk. The VAE can decode these video latents into future RGB observations.

4 Experiments

We first introduce the experimental setup (Sec. 4.1) and evaluate SpatialWAM against strong baselines on simulation benchmarks and real-world tasks (Sec. 4.2). We then assess geometric supervision, shared 3D scenes, and pre-training data through ablations (Sec. 4.3). Finally, we examine hyperparameter sensitivity, post-training data scaling, and future video prediction quality (Sec. 4.4).

4.1 Experimental Setup

RoboCasa Simulation Benchmark. RoboCasa (Nasiriany et al., 2024) evaluates kitchen manipulation with a Franka Panda robot. We use the 24-task benchmark and 50 training demonstrations per task. Following several prior works (Kim et al., 2026; Qian et al., 2026; Liang et al., 2025), we evaluate each task in five kitchen scenes with different layouts and styles, running 10 trials per scene. *Evaluation uses unseen object instances, and two scenes per task also have unseen styles.* We average success rates across tasks and three random seeds, totaling 3,600 evaluation trials.

LIBERO Simulation Benchmark. LIBERO (Liu et al., 2023) assesses language-conditioned manipulation across spatial arrangements, object identities, task goals, and long-horizon behaviors. Following Kim et al. (2025b;a; 2026), we use four suites: LIBERO-Spatial, LIBERO-Object, LIBERO-Goal, and LIBERO-Long, each containing 10 tasks with 50 training demonstrations per task. We train a separate policy for each suite and evaluate it over 50 trials per task. We report the average success rate across the four suites and three random seeds, totaling 6,000 evaluation trials.

LIBERO-Plus Simulation Benchmark. LIBERO-Plus (Fei et al., 2026) evaluates robustness under seven perturbation categories: camera viewpoints, robot initial states, language instructions, lighting, background textures, sensor noise, and object layouts. We evaluate the policies trained on the original LIBERO demonstrations without further adaptation, using the released set of 10,030 perturbed task instances. Each instance is evaluated once per seed. We report the average success rate across all instances and three random seeds, totaling 30,090 evaluation trials.

Real-World Evaluation. Our platform consists of a single robot arm, two external cameras, and a wrist-mounted camera, as shown in Figure 3(b). We evaluate three manipulation tasks shown in Figure 3(a), using 100 demonstration trajectories per task for training.
(1) **Remote Placement.** The robot places a remote control into a basket and then pushes it to the left or right edge as instructed. The basket's width only slightly exceeds the remote's length, leaving little clearance for placement. This task tests precise spatial alignment and language following.

Table 1: **RoboCasa Simulation Results.** We report success rates (%). Demos/Task denotes the number of training demonstrations per task. * denotes results reproduced by us. See Appendix C.1 for detailed results.

Method	Demos/Task	Average
UVA (Li et al., 2025)	50	50.0
GeoPredict (Qian et al., 2026)	50	52.4
DP-VLA (Han et al., 2024)	3000	57.3
UWM (Zhu et al., 2025)	1000	60.8
π_0 (Black et al., 2025b)	300	62.5
$\pi^*_{0.5}$ (Black et al., 2025a)	50	62.6
GR00T-N1.5 (NVIDIA et al., 2025)	300	64.1
Video Policy (Liang et al., 2025)	300	66.0
FLARE (Zheng et al., 2025)	300	66.4
Cosmos Policy (Kim et al., 2026)	50	67.1
SpatialWAM (ours)	50	**73.4**

Table 2: **LIBERO And LIBERO-Plus Simulation Results.** We report success rates (%). Bold and underlined values indicate the best and second-best results, respectively. See Appendices C.2 and C.3 for detailed results.

Method	LIBERO	LIBERO-Plus
OpenVLA (Kim et al., 2025b)	76.5	15.6
WorldVLA (Cen et al., 2025)	79.1	25.0
UniVLA (Bu et al., 2025)	95.2	42.9
Fast-WAM (Yuan et al., 2026)	97.6	50.0
π_0 (Black et al., 2025b)	94.2	53.6
RIPT-VLA (Tan et al., 2025)	97.5	68.4
OpenVLA-OFT (Kim et al., 2025a)	97.1	69.6
ELAN4D (He et al., 2026)	97.0	78.2
Cosmos Policy (Kim et al., 2026)	**98.5**	82.4
$\pi_{0.5}$ (Black et al., 2025a)	96.9	<u>84.6</u>
SpatialWAM (ours)	<u>97.8</u>	**85.9**

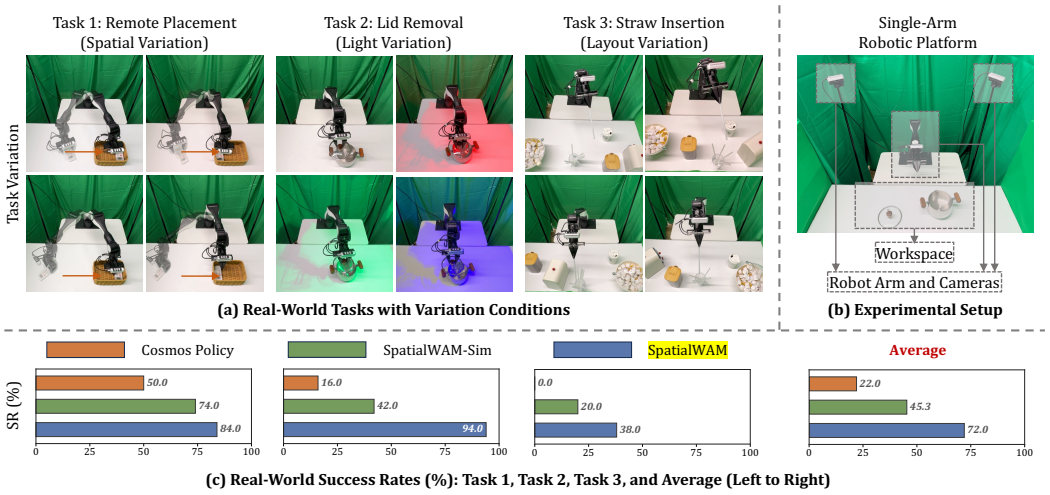

Figure 3: **Real-World Experimental Setup and Results.** We show task variations, the experimental setup, and success rates. Average denotes the mean success rate across the three tasks.

(2) **Lid Removal.** The robot removes a transparent lid from a stainless-steel pot and places it on the table under different lighting conditions. The reflective pot and transparent lid exhibit substantial appearance changes with lighting, testing robustness to visual variations during manipulation.
(3) **Straw Insertion.** The robot picks up a straw from a holder and inserts it into a cup under varying object layouts. Grasping the slender straw and aligning it with the cup opening require precise control, testing fine-grained spatial reasoning and manipulation accuracy.

Each method is evaluated over 50 trials per task, totaling 450 trials across the three methods and three tasks. We report per-task success rates and their average across the three tasks.

Baselines. We compare with vision–language–action models and world action models, including $\pi_{0.5}$ (Black et al., 2025a), OpenVLA-OFT (Kim et al., 2025a), Cosmos Policy (Kim et al., 2026), and Fast-WAM (Yuan et al., 2026), as well as geometry-aware policies GeoPredict (Qian et al., 2026) and ELAN4D (He et al., 2026). Tables 1 and 2 list simulation baselines. We use results from original papers and published benchmark evaluations (Fei et al., 2026), except for $\pi_{0.5}$ on RoboCasa, which we reproduce with 50 demonstrations per task. Real-world baselines are Cosmos Policy and SpatialWAM-Sim. SpatialWAM-Sim uses only simulation data for video–spatial pretraining, followed by post-training on the same real-world demonstrations as SpatialWAM.

Implementation Details. We restrict geometric supervision to a $1.6\,\mathrm{m} \times 1.6\,\mathrm{m} \times 1.0\,\mathrm{m}$ robot workspace. A voxel size of $v = 0.04\,\mathrm{m}$ yields a $40 \times 40 \times 25$ grid, with 4 Gaussian primitives predicted per voxel. Spatial queries form an $8 \times 8 \times 5$ grid, 5 times coarser along each axis. We use an action chunk length of $H = 32$ and a temporal downsampling factor of $r = 4$, predicting 8 future

Table 3: **Ablation Study on RoboCasa.** Average SR is the success rate (%) averaged across all 24 evaluation tasks.

Variant	Average SR
w/o Pre-Training	57.9
Video-Only Pre-Training	68.1
2D Depth Pre-Training	69.3
Simulation-Only Pre-Training	70.5
SpatialWAM	**73.4**

Table 4: **Hyperparameter Study on RoboCasa.** All other experimental settings match the SpatialWAM configuration.

Setting	Average SR
$H = 16,\ r = 2$	69.2
$H = 64,\ r = 8$	63.9
$d_a = 512$	70.8
$d_a = 2{,}048$	71.1
SpatialWAM	**73.4**

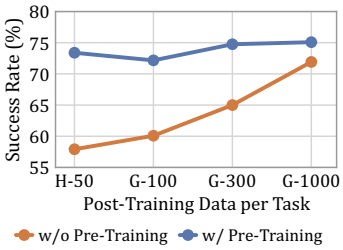

Figure 4: Data scaling study on RoboCasa's multiple settings.

RGB frames and 9 Gaussian scenes, including the current scene. We use 9 dynamics queries, 1 per geometric prediction time. The spatial expert uses 30 transformer layers with a hidden dimension of 1,024. Details of video–spatial pre-training and post-training are provided in Appendices A and B.

4.2 Main Results

RoboCasa. As shown in Table 1, SpatialWAM achieves a success rate of 73.4%, outperforming all compared methods. It improves over the strongest baseline, Cosmos Policy (Kim et al., 2026) (67.1%), by **6.3** percentage points. Using 50 Demos/Task, SpatialWAM also outperforms FLARE (Zheng et al., 2025), UWM (Zhu et al., 2025), and DP-VLA (Han et al., 2024), which use 300, 1,000, and 3,000 Demos/Task, respectively. These results support combining explicit 3D geometric learning with video representations for object and scene generalization.

LIBERO. SpatialWAM achieves 97.8% on LIBERO, ranking second among the methods in Table 2. It outperforms Fast-WAM (Yuan et al., 2026), OpenVLA-OFT (Kim et al., 2025a), and $\pi_{0.5}$ (Black et al., 2025a), while remaining within 0.7 percentage points of Cosmos Policy (Kim et al., 2026) (98.5%). Success rates exceed 95% across 4 suites (Appendix C.2), covering variations in spatial arrangements, object identities, task goals, and long-horizon behaviors. These results demonstrate competitive performance across language-conditioned manipulation tasks.

LIBERO-Plus. Without further adaptation, SpatialWAM achieves the highest success rate of 85.9% among the compared methods on LIBERO-Plus (Table 2). It exceeds $\pi_{0.5}$ (Black et al., 2025a), Cosmos Policy (Kim et al., 2026), and the geometry-aware ELAN4D (He et al., 2026) by 1.3, 3.5, and 7.7 percentage points, respectively. Although Cosmos Policy performs better on the original LIBERO tasks, SpatialWAM performs better under perturbations. This supports the value of explicit 3D geometric learning for robust robotic manipulation.

Real-World Experiments. Figure 3(c) shows that SpatialWAM achieves an average success rate of 72.0%, exceeding Cosmos Policy (Kim et al., 2026) (22.0%) by **50.0** percentage points. On Straw Insertion, SpatialWAM succeeds in 19 of 50 trials, while Cosmos Policy fails in all 50, highlighting gains on a task requiring precise grasping and spatial alignment. With identical post-training demonstrations, SpatialWAM outperforms SpatialWAM-Sim (45.3%) by 26.7 percentage points, suggesting that noisy real-world data in video–spatial pre-training yields representations more robust to real-world variability. Together, these results demonstrate SpatialWAM's effectiveness across tasks requiring precise spatial alignment and robustness to visual variations.

4.3 Ablation Study

Setting. Table 3 compares four pre-training variants with SpatialWAM on RoboCasa. *w/o Pre-Training* skips video–spatial pre-training, initializing the video expert from Wan2.2-5B (Wan et al., 2025) and randomly initializing the spatial and action experts and STGM. *Video-Only Pre-Training* pre-trains only the video expert on the same corpus used for video–spatial pre-training, then initializes post-training from these video weights with randomly initialized spatial and action experts. *2D Depth Pre-Training* jointly pre-trains the video expert and a depth expert on the same corpus. The depth expert uses 30 transformer layers with hidden dimension 1,024 and predicts multi-view depth maps at 9 time steps instead of shared 3D Gaussian scenes. During post-training, the video and depth experts use pre-trained weights, while a randomly initialized action expert accesses both experts' features through joint attention. Depth supervision is retained. *Simulation-Only Pre-Training* performs video–spatial pre-training using only the simulation subset of the pre-training corpus we construct. All variants use the same post-training demonstrations as SpatialWAM.

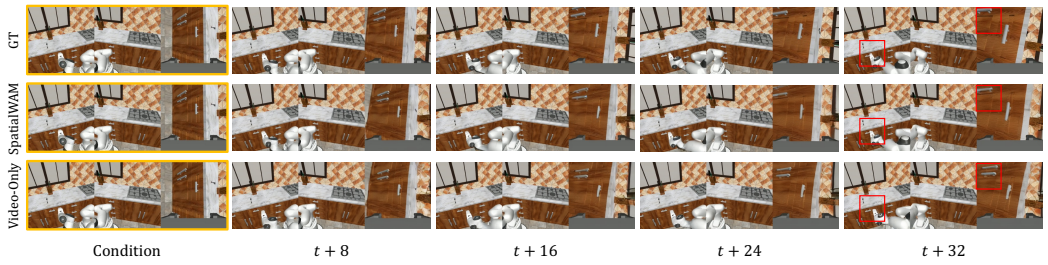

Figure 5: **Qualitative Comparison of Video Predictions.** Rows show ground truth (GT), SpatialWAM, and Video-Only Pre-Training. The first column shows the conditioning observation, followed by frames at $t+8$, $t+16$, $t+24$, and $t+32$. Red boxes highlight regions for comparison.

Results. SpatialWAM achieves 73.4%, outperforming all four variants in Table 3. The 15.5-point gain over *w/o Pre-Training* highlights the benefit of learning video and spatial representations before action post-training. With the same pre-training data, SpatialWAM exceeds *Video-Only Pre-Training* by **5.3** percentage points, supporting our design of using geometric learning to shape spatially grounded video representations for action generation. Its 4.1-point gain over *2D Depth Pre-Training* further underscores the value of explicit 3D scene representations shared across camera views. Finally, the 2.9-point gain over *Simulation-Only Pre-Training* suggests that real-world data provides complementary information for learning transferable spatial representations.

4.4 Further Analysis

Hyperparameter Sensitivity. Table 4 examines the action chunk length H and the action expert's hidden dimension d_a on RoboCasa. SpatialWAM uses $H = 32$, $r = 4$, and $d_a = 1{,}024$. For $H = 16$ and $H = 64$, we set $r = 2$ and $r = 8$, respectively, retaining 8 future RGB frames and 9 Gaussian scenes, including the current scene. These settings achieve 69.2% and 63.9% success, respectively, compared with 73.4% for SpatialWAM. Changing d_a to 512 or 2,048 yields 70.8% and 71.1%, respectively. The default configuration performs best among the tested settings, indicating that larger action chunks or wider action experts do not necessarily improve performance.

Data Scaling. Figure 4 compares SpatialWAM with *w/o Pre-Training* under four post-training data settings on RoboCasa. H-50 uses 50 human expert demonstrations per task, while G-100, G-300, and G-1000 use 100, 300, and 1,000 demonstrations generated by MimicGen (Mandlekar et al., 2023), respectively. Both models use identical post-training data per setting. Video–spatial pre-training improves success across four settings. Within the generated data, both models benefit from more demonstrations, while the gap narrows from 12.1 percentage points at G-100 to 3.2 at G-1000. With 100 generated demonstrations per task, SpatialWAM reaches 72.2%, comparable to 71.9% for *w/o Pre-Training* with 1,000 demonstrations. This suggests that video and spatial representations learned through video–spatial pre-training improve post-training data efficiency.

Qualitative Analysis. Figure 5 compares future video predictions from SpatialWAM and *Video-Only Pre-Training* given the same multi-view observation. In the highlighted regions at $t+32$, the gripper grasps the handle in both the ground truth and SpatialWAM's prediction, but fails to do so in the video-only prediction. The video-only variant also generates an additional cabinet handle, whereas SpatialWAM preserves the cabinet structure. These observations suggest that geometric supervision helps capture scene structure and gripper–object interactions during video prediction, providing qualitative evidence consistent with the manipulation gains in Table 3.

5 Conclusion

We presented SpatialWAM, which couples video models' spatio-temporal priors with explicit 3D geometric learning to obtain spatially grounded video and spatial representations for action generation. STGM fuses features learned by spatial and dynamics queries to predict shared 3D Gaussian scenes, whose multi-view depth supervision shapes both experts through bidirectional interaction. Action tokens access both spatially grounded video representations and spatial expert features through joint attention, including at inference when STGM is omitted. Simulation and real-world evaluations show strong manipulation performance. Ablations support geometric representation learning and shared 3D scene modeling. Further analyses suggest improved post-training data efficiency and more faithful video predictions. Pre-training without action annotations offers a path to extending this geometric representation learning to a broader range of calibrated RGB-D videos.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on research methodology and experiments and assist with literature translation. We did not use these tools to propose or refine hypotheses, implement the research method, clean or reformat datasets, support qualitative and thematic data analysis, or interpret results. Synthetic dataset generation, theoretical or conceptual framework development, and mathematical claims and proofs are not applicable to this work. Additionally, these tools supported literature search and summarization, information retrieval, manuscript drafting and language editing, table and reference formatting, and non-core code development. The authors manually reviewed all AI-assisted content, including manuscript text, code, and cited references. We accept full responsibility for the final manuscript and associated artifacts, including content produced with AI assistance.

ETHICS STATEMENT

This work studies robotic manipulation in simulation and real-world environments. We have identified no specific ethical concerns arising from the methods and experiments presented in this paper.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, Section 3.1 defines the task and video backbone, Section 3.2 details the spatial and dynamics queries and STGM, and Sections 3.3 and 3.4 describe video–spatial pre-training, action post-training, and inference, including expert interactions and training objectives. Section 4.1 specifies the benchmarks, real-world tasks, evaluation protocols, baselines, and key implementation settings. Appendix A provides pre-training data sources, preprocessing, and training configurations, while Appendix B reports model initialization, post-training configurations, and inference details. These appendices specify architecture parameters, optimizer settings, training steps, batch sizes, and computing resources. Sections 4.3 and 4.4 describe the ablation and analysis settings, and Appendix C reports detailed simulation results. We commit to releasing our source code upon acceptance of this paper.

REFERENCES

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, Hongyan Zhao, Hanyu Liu, Zhizhong Su, Lei Ma, Hang Su, and Jun Zhu. Motus: A unified latent action world model. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35101–35113, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Bi_Motus_A_Unified_Latent_Action_World_Model_CVPR_2026_paper.html.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40, 2025a. URL https://proceedings.mlr.press/v305/black25a.html.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems*, 2025b. doi: 10.15607/RSS.2025.XXI.010. URL https://www.roboticsproceedings.org/rss21/p010.html.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Learning to act anywhere with task-centric latent actions. In *Robotics:*

Science and Systems, 2025. doi: 10.15607/RSS.2025.XXI.014. URL https://www.roboticsproceedings.org/rss21/p014.html.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, Deli Zhao, and Hao Chen. WorldVLA: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025. URL https://arxiv.org/abs/2506.21539.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: A progressive robustness benchmark for visual-language-action models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 38574–38583, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Fei_LIBERO-Plus_A_Progressive_Robustness_Benchmark_for_Visual-Language-Action_Models_CVPR_2026_paper.html.

Jun Guo, Qiwei Li, Peiyan Li, Zilong Chen, Nan Sun, Yifei Su, Heyun Wang, Yuan Zhang, Xinghang Li, and Huaping Liu. Unified 4D world action modeling from video priors with asynchronous denoising. *arXiv preprint arXiv:2604.26694*, 2026. URL https://arxiv.org/abs/2604.26694.

ByungOk Han, Jaehong Kim, and Jinhyeok Jang. A dual process vla: Efficient robotic manipulation leveraging vlm. *arXiv preprint arXiv:2410.15549*, 2024. URL https://arxiv.org/abs/2410.15549.

Zeyuan He, Bowen Yang, Zhirui Fang, Keru Zhou, Lei Jiang, Jingjing Qian, Fan Mo, Junchi Yan, Philip Torr, Xiu Li, Li Jiang, and Jialin Yu. ELAN4D: Embodiment-centric 4D supervision for vision-language-action models via plug-and-play adaptation. *arXiv preprint arXiv:2605.30484*, 2026. URL https://arxiv.org/abs/2605.30484.

Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Fei-Fei Li. PointWorld: Scaling 3D world models for in-the-wild robotic manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20765–20779, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/papers/Huang_PointWorld_Scaling_3D_World_Models_for_In-The-Wild_Robotic_Manipulation_CVPR_2026_paper.pdf.

Jiahao Jiang, Jianing Zhang, Zhenhan Yin, Ruidong Chen, Sen Wang, Zhaoshu Yu, Pengpeng Zeng, Xiaofeng Cao, Xuanhan Wang, Jingkuan Song, and Heng Tao Shen. WSA_1: a 3D-centric world-spatial-action model for generalizable robot control. *arXiv preprint arXiv:2607.03941*, 2026. URL https://arxiv.org/abs/2607.03941.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023. URL https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Donovon Jackson, Charlotte Le, Yunshuang Li, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu,

Thomas Kollar, Sergey Levine, and Chelsea Finn. DROID: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.120. URL https://www.roboticsproceedings.org/rss20/p120.html.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Robotics: Science and Systems*, 2025a. doi: 10.15607/RSS.2025.XXI.017. URL https://www.roboticsproceedings.org/rss21/p017.html.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P. Foster, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713, 2025b. URL https://proceedings.mlr.press/v270/kim25c.html.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-tuning video models for visuomotor control and planning. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/748becc400a57c0e31cfe6a2e7951467-Abstract-Conference.html.

Chengmeng Li, Junjie Wen, Yaxin Peng, Yan Peng, and Yichen Zhu. PointVLA: Injecting the 3D world into vision-language-action models. *IEEE Robotics and Automation Letters*, 11(3):2506–2513, 2026a. doi: 10.1109/LRA.2026.3653303. URL https://ieeexplore.ieee.org/document/11346992/.

Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. In *International Conference on Learning Representations*, 2026b. URL https://openreview.net/forum?id=euMVC1DO4k.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Luyao Zhang, Mingrui Yu, Zelin Gao, Nan Xue, Boyu Zhou, Xing Zhu, Mingyu Ding, Yujun Shen, and Yinghao Xu. Causal world modeling for robot control. In *Robotics: Science and Systems*, 2026c. doi: 10.15607/RSS.2026.XXII.016. URL https://www.roboticsproceedings.org/rss22/p016.html.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. In *Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.074. URL https://www.roboticsproceedings.org/rss21/p074.html.

Ying Li, Xiaobao Wei, Jiajun Cao, Hao Wang, Xiaowei Chi, Chengyu Bai, Qianpu Sun, Jiajun Li, Xiaojie Zhang, Peidong Jia, Jian Tang, Sirui Han, and Shanghang Zhang. WAM4D: Fast 4D world action model via spatial register tokens. *arXiv preprint arXiv:2606.14048*, 2026d. URL https://arxiv.org/abs/2606.14048.

Junbang Liang, Pavel Tokmakov, Ruoshi Liu, Sruthi Sudhakar, Paarth Shah, Rares Ambrus, and Carl Vondrick. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*, 2025. URL https://arxiv.org/abs/2508.00795.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8c3c666820ea055a77726d66fc7d447f-Abstract-Datasets_and_Benchmarks.html.

Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. MimicGen: A data generation system for scalable robot learning using human demonstrations. In *Conference on Robot Learning*, 2023. URL https://mimicgen.github.io/.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of household tasks for generalist robots. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.050. URL https://www.roboticsproceedings.org/rss20/p050.html.

Soroush Nasiriany, Sepehr Nasiriany, Abhiram Maddukuri, and Yuke Zhu. RoboCasa365: A large-scale simulation framework for training and benchmarking generalist robots. In *International Conference on Learning Representations*, 2026. URL https://robocasa.ai/assets/robocasa365_iclr26.pdf.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. In *ArXiv Preprint*, March 2025. URL https://arxiv.org/abs/2503.14734.

Jingjing Qian, Boyao Han, Chen Shi, Lei Xiao, Long Yang, Shaoshuai Shi, and Li Jiang. GeoPredict: Leveraging predictive kinematics and 3D gaussian geometry for precise VLA manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13529–13539, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Qian_GeoPredict_Leveraging_Predictive_Kinematics_and_3D_Gaussian_Geometry_for_Precise_CVPR_2026_paper.html.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Jiayuan Gu, Zhigang Wang, Yan Ding, Bin Zhao, Dong Wang, and Xuelong Li. SpatialVLA: Exploring spatial representations for visual-language-action models. In *Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.011. URL https://www.roboticsproceedings.org/rss21/p011.html.

Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. VideoVLA: Video generators can be generalizable robot manipulators. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/89a3b655a8b68ae1c76b768152c9c19d-Abstract-Conference.html.

Shuhan Tan, Kairan Dou, Yue Zhao, and Philipp Krähenbühl. Interactive post-training for vision-language-action models. *arXiv preprint arXiv:2505.17016*, 2025. URL https://arxiv.org/abs/2505.17016.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. URL https://arxiv.org/abs/2503.20314.

Lei Xiao, Jifeng Li, Juntao Gao, Feiyang Ye, Yan Jin, Jingjing Qian, Jing Zhang, Yong Wu, and Xiaoyuan Yu. AVA-VLA: Improving vision-language-action models with active visual attention. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13453–13463, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Xiao_AVA-VLA_Improving_Vision-Language-Action_models_with_Active_Visual_Attention_CVPR_2026_paper.html.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William

Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026. URL https://arxiv.org/abs/2602.15922.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026. URL https://arxiv.org/abs/2603.16666.

Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, XinQiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, Li Yi, Wenjun Zeng, and Xin Jin. DreamVLA: A vision-language-action model dreamed with comprehensive world knowledge. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-0818. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/22d4f952efa13970f0b1ffb22170d416-Abstract-Conference.html.

Ruijie Zheng, Jing Wang, Scott Reed, Johan Bjorck, Yu Fang, Fengyuan Hu, Joel Jang, Kaushil Kundalia, Zongyu Lin, Loïc Magne, Avnish Narayan, You Liang Tan, Guanzhi Wang, Qi Wang, Jiannan Xiang, Yinzhen Xu, Seonghyeon Ye, Jan Kautz, Furong Huang, Yuke Zhu, and Linxi Fan. FLARE: Robot learning with implicit world modeling. In *Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 3952–3971, 2025. URL https://proceedings.mlr.press/v305/zheng25a.html.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.015. URL https://www.roboticsproceedings.org/rss21/p015.html.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: Vision-language-action models transfer web knowledge to robotic control. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

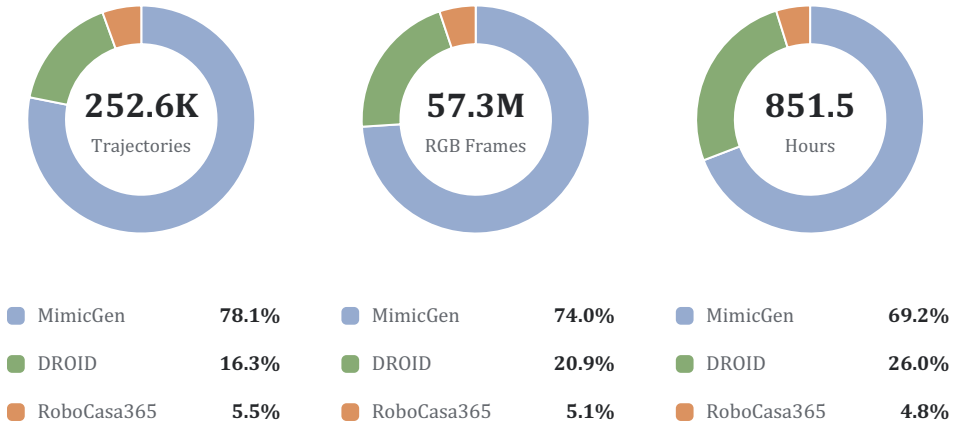

Figure 6: **Composition of the Pre-Training Data.** The three rings show each source's share of trajectories (left), RGB frames (middle), and duration (right). Center labels report rounded totals, and legends show the corresponding percentages.

Table 5: **Pre-Training Data Statistics by Source.** Durations are computed using the respective frame rates and rounded to two decimal places.

Source	Trajectories	RGB Frames	FPS	Hours
MimicGen (Mandlekar et al., 2023)	197,385	42,414,916	20	589.10
DROID (Khazatsky et al., 2024)	41,298	11,957,897	15	221.44
RoboCasa365 (Nasiriany et al., 2026)	13,966	2,951,936	20	41.00
Total	**252,649**	**57,324,749**	–	**851.54**

A Pre-Training Details

Data. Our pre-training corpus contains 252,649 single-arm trajectories from MimicGen (Mandlekar et al., 2023), RoboCasa365 (Nasiriany et al., 2026), and DROID (Khazatsky et al., 2024), comprising 57,324,749 RGB frames and 851.54 hours. Figure 6 shows each source's contribution by trajectory count, frame count, and duration, and Table 5 reports the detailed statistics. We retain the original frame rates of 20 fps for MimicGen and RoboCasa365 and 15 fps for DROID, without temporal resampling. We compute each source's duration from its frame count and frame rate before summing across sources. RGB observations from the left external, right external, and wrist cameras are concatenated into 224×672 inputs. Depth supervision uses the left and right external views at 224×224. We standardize camera intrinsics and extrinsics across data sources, with all camera poses expressed in a common workspace coordinate system.

Training Setup. We initialize the video expert from Wan2.2-5B (Wan et al., 2025) and randomly initialize the spatial expert. The spatial expert uses 30 transformer layers with a hidden dimension of 1,024. We predict 8 future RGB frames and 9 Gaussian scenes, including the current scene. Each RGB frame concatenates the left external, right external, and wrist camera views into an image with a height of 224 pixels and a width of 672 pixels. The spatial expert uses $8 \times 8 \times 5 = 320$ spatial queries and 9 dynamics queries. The 3D Gaussian Decoder outputs a $40 \times 40 \times 25$ voxel grid, with 4 Gaussian primitives predicted per voxel. Geometric supervision uses 224×224 depth maps from the 2 external cameras. Both experts and STGM are fully updated using the joint objective in Eq. 7, while the video VAE remains frozen. Language conditioning uses precomputed T5 embeddings, and the language encoder is not updated during pre-training. We set $\lambda_{\text{video}} = \lambda_{\text{depth}} = 1$. We use Adam with a learning rate of 5.0×10^{-5}. We pre-train for 110,000 steps with a global batch size of 512 on 64 NVIDIA H20 GPUs, taking approximately 140 hours.

Table 6: **Task-Wise RoboCasa Results** (Nasiriany et al., 2024). We report SpatialWAM's success rates (%) averaged over 3 random seeds for each task and across all 24 tasks.

Task	SR	Task	SR	Task	SR	Task	SR
PnPCabToCounter	52.0	PnPSinkToCounter	64.0	CloseDoubleDoor	83.3	TurnOffSinkFaucet	90.0
PnPCounterToCab	63.3	PnPStoveToCounter	74.0	CloseSingleDoor	100.0	TurnSinkSpout	72.7
PnPCounterToMicrowave	34.0	OpenSingleDoor	91.3	OpenDrawer	91.3	CoffeePressButton	94.7
PnPCounterToSink	55.3	OpenDoubleDoor	100.0	CloseDrawer	99.3	TurnOnMicrowave	90.0
PnPCounterToStove	62.7	TurnOnStove	80.0	TurnOnSinkFaucet	80.7	TurnOffMicrowave	98.7
PnPMicrowaveToCounter	35.3	TurnOffStove	34.0	CoffeeServeMug	77.3	CoffeeSetupMug	37.3
						Overall	**73.4**

B Post-Training Details

Initialization and Architecture. We initialize the video and spatial experts and STGM from our video–spatial pre-training checkpoint and randomly initialize the action expert. The spatial and action experts each use 30 transformer layers with a hidden dimension of 1,024. The spatial expert retains 320 spatial queries and 9 dynamics queries. The 3D Gaussian Decoder outputs a $40 \times 40 \times 25$ voxel grid, with 4 Gaussian primitives predicted per voxel.

Training Setup. We use an action chunk length of $H = 32$ and a temporal downsampling factor of $r = 4$, predicting 8 future RGB frames and 9 Gaussian scenes, including the current scene, while retaining all 32 actions. The action expert processes 1 proprioceptive state token and 32 action tokens. For RoboCasa, LIBERO, and all real-world tasks, we concatenate RGB images from 2 external cameras and 1 wrist-mounted camera into frames with a height of 224 pixels and a width of 672 pixels. Geometric supervision uses 224×224 depth maps from the 2 external cameras. All 3 experts and STGM are fully updated using the joint objective in Eq. 8, while the video VAE remains frozen. Language conditioning uses precomputed T5 embeddings, and the language encoder is not updated during post-training. We set $\lambda_{\text{video}} = \lambda_{\text{depth}} = \lambda_{\text{action}} = 1$. We use Adam with a learning rate of 5.0×10^{-5} for all benchmarks. For RoboCasa and LIBERO, we post-train for 40,000 steps with a global batch size of 256 on 32 NVIDIA H20 GPUs. For all real-world tasks, we post-train for 8,000 steps with a global batch size of 128 on 16 NVIDIA H20 GPUs.

Inference. At inference, all 3 experts are retained and STGM is omitted. Action tokens access both spatially grounded video representations and spatial expert features through joint attention during post-training and inference. The model jointly denoises future video latents and the 32-step action chunk through joint attention with 320 spatial queries and 9 dynamics queries, conditioned on the current observations, proprioceptive state, and language instruction.

C Detailed Simulation Results

C.1 RoboCasa

Table 6 reports SpatialWAM's performance across 24 RoboCasa tasks, with an overall success rate of 73.4%. SpatialWAM achieves 100.0% on OpenDoubleDoor and CloseSingleDoor and 99.3% on CloseDrawer, demonstrating reliable performance on these articulated-object manipulation tasks. The 8 pick-and-place tasks average 55.1%, with PnPCounterToMicrowave and PnPMicrowaveToCounter reaching 34.0% and 35.3%, respectively, identifying object transfer involving the microwave as a particularly challenging setting. CoffeeSetupMug and TurnOffStove also remain difficult, with success rates of 37.3% and 34.0%, highlighting variation in performance across kitchen manipulation tasks.

C.2 LIBERO

The Original column in Table 7 reports SpatialWAM's performance across the 4 LIBERO suites, with an overall success rate of 97.8%. SpatialWAM achieves 99.1% on both LIBERO-Spatial and

Table 7: **Detailed LIBERO and LIBERO-Plus Results** (Liu et al., 2023; Fei et al., 2026). We report SpatialWAM's success rates (%) averaged over 3 random seeds. Columns correspond to original LIBERO and LIBERO-Plus perturbations in camera viewpoints, robot initial states, language instructions, lighting conditions, background textures, sensor noise, and object layouts.

Suite	Original	Camera	Robot	Language	Lighting	Background	Noise	Layout
Spatial	99.1	90.8	78.7	95.0	99.1	98.6	95.4	93.2
Object	99.1	90.7	75.9	99.2	99.1	98.5	98.7	78.9
Goal	97.5	93.9	66.7	67.1	98.9	90.7	96.3	63.8
Long	95.5	61.8	62.1	94.3	90.5	87.2	88.9	78.5

LIBERO-Object, followed by 97.5% on LIBERO-Goal and 95.5% on LIBERO-Long. These results show consistently high performance across spatial arrangements, object identities, and task goals. LIBERO-Long yields the lowest success rate, indicating that long-horizon manipulation remains relatively more challenging, although all 4 suites exceed 95%.

C.3 LIBERO-PLUS

Table 7 reports SpatialWAM's performance across the 7 LIBERO-Plus perturbation categories, with an overall success rate of 85.9% without further adaptation. Success rates remain at least 87.2% across all 4 suites under lighting, background texture, and sensor noise perturbations, indicating robustness to these visual variations. Camera viewpoint and robot initial-state perturbations are particularly challenging for LIBERO-Long, reducing its success rate from 95.5% to 61.8% and 62.1%, respectively. LIBERO-Goal also exhibits pronounced drops under language and object layout perturbations, reaching 67.1% and 63.8%, compared with 97.5% on the original tasks. These results reveal distinct sensitivities across suites and identify settings where strong performance on unperturbed tasks does not translate into comparable robustness.

D REAL-WORLD EXECUTION VIDEOS

The `videos` directory of the supplementary materials contains real-world execution videos of SpatialWAM and Cosmos Policy (Kim et al., 2026). The videos cover the three tasks described in Section 4.1: Remote Placement, Lid Removal, and Straw Insertion. For each task, we test both methods under four variations and provide the corresponding execution results in the videos.