SCOPEVLA: TASK-GROUNDED AND VIEW-SPECIALIZED SPATIAL CONDITIONING FOR ROBOTIC MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Incorporating geometric information into vision-language-action (VLA) models has become an important approach to improving robotic manipulation performance. However, identifying task-relevant spatial evidence and exploiting the distinct roles of third-person and wrist views remain challenging. In this paper, we introduce ScopeVLA, a task-grounded framework for view-specialized spatial conditioning. ScopeVLA employs Global and Wrist queries to extract task-relevant spatial distributions from intermediate vision-language model (VLM) representations. These queries are supervised by view-specific workspace targets derived from end-effector trajectories. These distributions guide two complementary conditioning pathways: the Global prior introduces a confidence-aware attention bias over third-person visual tokens, while the Wrist query assignments spatially modulate multi-level features extracted from a frozen geometry model. The modulated geometric features are projected into additional keys and values for joint attention in the action expert. Both pathways operate at selected action-expert layers while preserving the original layerwise vision-language conditioning. Controlled instruction interventions show that the learned spatial distributions remain stable under task-preserving paraphrases while responding to changes in task instructions. Experiments on two simulation benchmarks and real-world manipulation tasks show that ScopeVLA improves task success over the evaluated VLA baselines.

1 INTRODUCTION

Vision-language-action (VLA) models provide a promising foundation for general-purpose robotic manipulation by translating visual observations and language instructions into executable actions (Kim et al., 2024; Black et al., 2026). Their success builds on the semantic knowledge acquired by pretrained vision-language models (VLMs), enabling robots to interpret instructions and transfer knowledge across tasks. Yet successful manipulation requires more than identifying *what* to manipulate. The robot must also resolve spatial relationships that determine *where* and *how* to approach and interact with the target. This has motivated growing interest in incorporating 3D representations and geometric priors into VLA policies (Zhen et al., 2024; Qu et al., 2025; Li et al., 2026a; Zhang et al., 2026b).

Existing approaches introduce geometry at different stages of the VLA pipeline. Input-level methods enrich VLM observations with spatial representations (Qu et al., 2025; Li et al., 2026c), while action-level methods provide the action decoder with point-cloud or geometry-aware features (Li et al., 2026a; Sun et al., 2025; Zhang et al., 2026b; Chen et al., 2026a). Other approaches transfer geometric knowledge through auxiliary supervision or representation alignment (Li et al., 2026b; Guo et al., 2025; Sun et al., 2026; Wang et al., 2026; Qian et al., 2026). Together, these studies demonstrate the value of geometric information for manipulation. However, dense geometric representations describe both task-relevant structures and the rest of the observed scene, leaving the policy to determine which spatial evidence actually matters for the current task. Effective geometric integration therefore requires not only richer spatial information, but also a mechanism for identifying and selectively exploiting task-relevant regions.

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.

This challenge becomes more pronounced under multi-view observations. Third-person cameras provide global scene layout and object relationships, whereas wrist cameras capture local gripper–object geometry and interaction details. Treating these views uniformly may overlook their complementary roles. Previous work has explored cross-view attention, perspective alignment, and wrist-focused prediction (Jangir et al., 2022; Bian et al., 2025; Fan et al., 2026; Pan et al., 2026; Zhao et al., 2026), but how task-relevant spatial evidence should be extracted from different views and incorporated into geometry-enhanced VLA policies remains less explored.

To this end, we propose ScopeVLA, a task-grounded framework for view-specialized spatial conditioning. ScopeVLA introduces learnable Global (G) and Wrist (W) queries that read intermediate VLM representations and produce task-conditioned spatial distributions for the third-person and wrist views. These queries are supervised by view-specific workspace targets derived from end-effector trajectories. An asymmetric attention mask keeps these grounding queries separate from the original prediction stream, allowing them to extract task-region information without altering the original VLM token-visibility structure. The Global distribution provides a confidence-aware prior over task-relevant third-person visual tokens, while the Wrist query responsibilities guide Wrist Region-Aware Spatial Feature Modulation (W-RSFM) over multi-level geometric features from a frozen geometry model. The modulated wrist features are incorporated into selected action-head layers as additional keys and values for joint attention, while preserving the original layerwise visual-language conditioning. A two-stage training scheme first learns the G/W grounding routes and then optimizes the manipulation policy.

Experiments on simulation benchmarks and real-world manipulation tasks validate both the learned grounding behavior and its contribution to policy learning. Controlled instruction and scene interventions show that the G/W routes respond to task semantics and scene-specific interaction regions rather than merely reproducing visual saliency. On CALVIN ABC→D, ScopeVLA improves the average completed sequence length from 4.50 to 4.62, with five-task success increasing from 80.0% to 85.3%. On LIBERO-Plus, it improves the overall zero-shot success rate from 59.1% to 67.4% over the strongest evaluated baseline. In real-world experiments, ScopeVLA further raises the average success rate from 61.25% to 80.0%. Ablation studies further verify the effectiveness of task-conditioned grounding and view-specialized spatial integration.

Our contributions are threefold:

- We introduce a trajectory-grounded dual-view readout that extracts task-relevant spatial distributions from intermediate VLM representations using Global and Wrist queries, providing view-specific workspace guidance without requiring manually annotated task-region masks.
- We propose a role-asymmetric spatial integration mechanism that uses Global grounding to guide third-person visual attention and Wrist grounding to selectively modulate local geometric features, integrating the two views according to their complementary roles in manipulation.
- We validate ScopeVLA across simulation and real-world manipulation settings, and show through controlled interventions and ablations that task-conditioned grounding, view-specialized geometric allocation, and selected-layer spatial conditioning contribute to improved policy performance.

2 Related Work

2.1 Vision-Language-Action Models

Vision-language-action (VLA) models translate visual observations and language instructions into robot actions. Following the scalable Transformer policies of RT-1 (Brohan et al., 2022), RT-2 (Brohan et al., 2023) and OpenVLA (Kim et al., 2024) adapt pretrained vision-language models (VLMs) through action token prediction. Subsequent methods, including CogACT (Li et al., 2024), π_0 (Black et al., 2026), and GR00T N1 (Bjorck et al., 2025), combine VLM representations with dedicated continuous action modules. OpenVLA-OFT (Kim et al., 2025) and SmolVLA (Shukor et al., 2025) further improve adaptation and inference efficiency. VLA-Adapter (Wang et al., 2025)

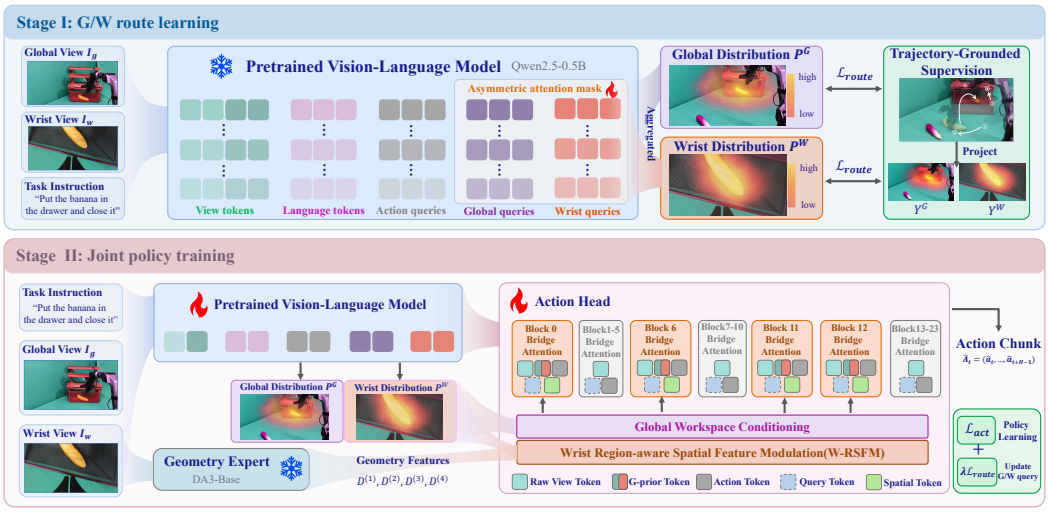

Figure 1: Overview of the ScopeVLA framework.

bridges raw multimodal and action-query representations to an action expert through layerwise conditioning.

2.2 Geometry for Vision-Language-Action Models

Existing methods incorporate geometry through spatial representations, action conditioning, or auxiliary supervision. SpatialVLA (Qu et al., 2025) and BridgeVLA (Li et al., 2026c) introduce spatial structure through positional encoding and aligned observation–action representations. PointVLA (Li et al., 2026a), GeoVLA (Sun et al., 2025), and FALCON (Zhang et al., 2026b) supply point-cloud or pretrained geometric features to action generation. Within this direction, PointVLA selects injection blocks, 3D-Mix (Yu et al., 2026) introduces semantic-conditioned fusion, and GeoAlign (Chen et al., 2026b) and VistaVLA (Liu et al., 2026) construct compact geometric conditions through state-guided querying and spatial summarization, respectively. Alternatively, Spatial Forcing (Li et al., 2026b), GLaD (Guo et al., 2025), ROCKET (Sun et al., 2026), and VEGA (Wang et al., 2026) transfer geometric knowledge through representation alignment, while GeoPredict (Qian et al., 2026) uses predictive geometric supervision.

2.3 View-Specific Processing for Robotic Manipulation

Prior work demonstrates the value of accounting for differences between camera views. Look Closer (Jangir et al., 2022) combines external scene context and egocentric detail through cross-view attention, while VLA-LPAF (Bian et al., 2025) aligns reference and auxiliary perspectives in latent space. More explicitly, World-to-Wrist (Pan et al., 2026) uses task context to condition future wrist representations, and GWM-VLA (Zhao et al., 2026) aggregates multi-view geometry while specializing future-state prediction to a selected wrist view. These studies support exploiting view-dependent information through differentiated processing.

3 Method

3.1 Overall Architecture

As illustrated in Figure 1, we propose ScopeVLA, a task-grounded framework for view-specialized spatial conditioning. Learnable Global (G) and Wrist (W) queries read intermediate VLM representations to produce view-specific, task-conditioned spatial distributions supervised by trajectory-derived workspace targets (Sec. 3.2). In parallel, a frozen geometry expert extracts multi-level spatial features from the wrist view. These signals are integrated into a VLA-Adapter-based action head according to their complementary roles (Sec. 3.3): the Global distribution guides attention over third-person visual tokens, while the Wrist query responsibilities drive W-RSFM to selectively

modulate wrist-view geometric features, which are introduced as additional keys and values at selected action-head layers. These representations predict an action chunk $\hat{\mathbf{A}}_t = (\hat{\mathbf{a}}_t, \ldots, \hat{\mathbf{a}}_{t+H-1})$, where H is the prediction horizon. In addition, ScopeVLA is trained in two stages, first learning the G/W grounding routes and then jointly optimizing the manipulation policy (Sec. 3.4).

3.2 View-specialized Task Region Grounding

Geometry foundation models produce dense spatial features that capture both manipulation-relevant structures and task-irrelevant scene content. In geometry-enhanced VLAs, these features are often provided to the policy as a whole, requiring task-dependent spatial relevance to be learned only indirectly through the action objective. Human manipulation, in contrast, is inherently task-directed: attention is selectively allocated to the target and its surrounding interaction region rather than distributed uniformly across all visible structures. Motivated by this observation, we investigate whether frozen VLM representations already encode task-conditioned spatial cues that can be explicitly elicited from both third-person and wrist views.

3.2.1 Is Task-Region Knowledge Accessible in Frozen VLMs?

To examine whether frozen VLM representations already encode task-conditioned spatial knowledge, we probe their native attention before introducing any trainable grounding queries or action components. Since manipulation-relevant regions extend beyond the target object to include interaction areas and motion paths, we use trajectory-derived interaction regions as spatial references. We then inspect visual attention across Qwen layers while holding the observations fixed and varying the instruction, including task-preserving paraphrases and different-task instructions, to evaluate both spatial localization and task sensitivity.

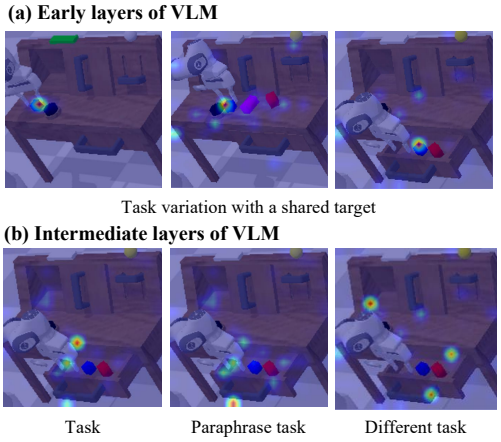

Figure 2: (a) Attention maps from the early-layer head L0H7 across different manipulation tasks sharing the blue block as the target. (b) Attention maps from the intermediate-layer head L11H1 under the original task instruction, a same-task paraphrase, and a different-task instruction.

The native attention exhibits complementary but incomplete properties. Early-layer heads show localized responses to objects and workspace structures, whereas selected intermediate-layer heads are more sensitive to instruction changes. We use LxHy to denote attention head y in the zero-indexed Qwen block x. As shown in Figure 2(a), early VLM layers (L0H7) exhibit pronounced object-centric sensitivity when the manipulation task varies while the target object remains fixed. Figure 2(b) further reveals greater instruction sensitivity in intermediate layers (L11H1). Their spatial attention remains largely stable under task-preserving paraphrases, but changes noticeably when the instruction specifies a different task. These observations suggest that workspace-aware and task-sensitive signals are present in the frozen VLM but may not be directly exposed as a coherent task-region distribution. We therefore investigate a lightweight, trajectory-supervised readout of intermediate VLM representations.

3.2.2 Trajectory-Grounded Role-Restricted G/W Queries

Trajectory-grounded supervision. We supervise task-region grounding with a continuous workspace field rather than manually annotated object masks. For each training episode, the end-effector trajectory defines a swept field F_τ in world coordinates. The field assigns pose-dependent strength and spatial extent along the trajectory, emphasizing the interaction region while retaining the approach path. At observation time t, we back-project valid depth pixels from the static (G) and wrist (W) cameras into the same world frame and evaluate F_τ on each view's observed surface. Robust aggregation within each visual patch yields two soft targets, Y_t^G and Y_t^W. Thus, the views

share a task-level workspace definition but receive distinct targets according to what each camera observes. Depth, calibration, and trajectory information are used to construct training targets; the G/W readout does not require them at inference.

Role-restricted queries and shared routes. We append K learnable Global queries and K learnable Wrist queries after the instruction tokens. Since the G/W queries serve as parallel grounding probes rather than causal action-prediction tokens, we use an asymmetric attention mask to prevent their route-supervised representations from interfering with the original prediction stream. To select the grounding readout layer, we compare candidate Qwen layers on CALVIN following the instruction-sensitive behavior observed in Section 3.2.1. Block 12 provides the best balance between third-person and wrist-view grounding, and is therefore used to produce the G/W spatial distributions supervised by the trajectory-derived targets.

For each view $v \in \{G, W\}$, lightweight query–key projections score the queries against the corresponding visual patches. The scores are across queries using log-mean-exp and normalized jointly with a learned NULL score, yielding a shared distribution $\mathbf{p}_t^v$ over N visual patches and the NULL option. Supervision from the trajectory-derived targets teaches this readout to identify the task workspace. The NULL option accounts for workspace relevance that cannot be assigned to the observed patches: without it, normalization would force all probability onto visible regions even when little or no relevant workspace is visible. This allows the readout to represent both where relevant evidence lies and how much is present, rather than producing an equally normalized spatial prediction for every observation.

Within each view, we further introduce competition among queries at each patch:

$$\omega_{t,i,j}^v = \mathrm{softmax}_i(z_{t,i,j}^v/\tau), \qquad \rho_{t,i,j}^v = p_{t,j}^v \omega_{t,i,j}^v, \qquad m_{t,i}^v = \sum_{j=1}^{N} \rho_{t,i,j}^v, \tag{1}$$

where $z_{t,i,j}^v$ is the query–patch score, $\tau > 0$ is the temperature, $\omega_{t,i,j}^v$ is the assignment probability across queries, $\rho_{t,i,j}^v$ is the spatial responsibility, and $m_{t,i}^v$ is the total mass assigned to query i. Together with the specialization regularizer introduced in Section 3.4, this competition encourages differentiated spatial responsibilities rather than redundant predictions across queries, while preserving a shared workspace distribution for each view.

3.3 Role-Asymmetric Spatial Integration

Section 3.2 produces task-conditioned spatial distributions for the third-person and wrist views. A straightforward design would use the same conditioning mechanism for both views. However, this symmetric treatment does not explicitly account for their distinct observational roles. We instead use the Global (G) distribution to regulate how the policy reads the existing third-person visual tokens, while the Wrist (W) responsibilities modulate dense geometric features extracted from the wrist observation. The two signals are integrated into the VLA-Adapter-based action head through complementary mechanisms.

3.3.1 Global Workspace Conditioning

At time step t, let $\mathbf{p}_t^G$ denote the Global distribution defined in Section 3.2.2. Its first N entries correspond to the third-person visual patches, while $p_{t,\emptyset}^G$ denotes the NULL probability. We redistribute the NULL mass uniformly across the visible patches to obtain the workspace prior

$$\pi_{t,j}^G = p_{t,j}^G + \frac{p_{t,\emptyset}^G}{N}, \qquad j = 1, \ldots, N. \tag{2}$$

This construction adaptively regulates the strength of spatial conditioning, allowing the workspace prior to revert smoothly to a uniform distribution when no reliable task-relevant spatial evidence is available, thereby avoiding spurious spatial bias.

As illustrated in Figure 3(a), we incorporate the workspace prior into Bridge Attention through a lightweight additive bias:

$$\widetilde{\mathbf{V}}_t^G = \sum_{j=1}^{N} \mathrm{Norm}_j\big(S_{t,j}^G + \tanh(\beta^G)\log\big(N\pi_{t,j}^G\big)\big)\,\mathbf{V}_{t,j}^G. \tag{3}$$

where $S^G_{t,j}$ is the original attention logit associated with the j-th Global-view visual value $\mathbf{V}^G_{t,j}$, and $\mathrm{Norm}_j(\cdot)$ denotes the original Bridge Attention normalization over visual tokens. The learnable parameter β^G controls the signed strength of the prior. We set $\beta^G = 0$ to preserve the original attention at initialization. Motivated by the layerwise attention analysis in Section 3.2.1 and the inherent characteristics of the VLA-Adapter action head, Global conditioning is introduced at $\mathcal{L}_G = \{0, 6, 11, 12\}$.

3.3.2 Wrist Region-aware Spatial Feature Modulation

As illustrated in Figure 3(b), the Wrist branch conditions dense geometric features rather than directly modifying attention over raw wrist-view tokens. For each Wrist query $i \in \{1, \ldots, K\}$, we use the responsibility map $\boldsymbol{\rho}^W_{t,i}$ and its mass $m^W_{t,i}$ defined in Section 3.2.2. Each responsibility map is linearly interpolated onto the geometry grid and normalized independently, while its original mass is retained. This query-wise treatment preserves the specialization induced by competitive assignment rather than collapsing all queries into the shared Wrist distribution. Specifically, independent normalization retains the spatial pattern captured by each query, while $m^W_{t,i}$ determines its contribution to the aggregated modulation. Complementary spatial roles can therefore be integrated according to their relevance without discarding their distinct response patterns.

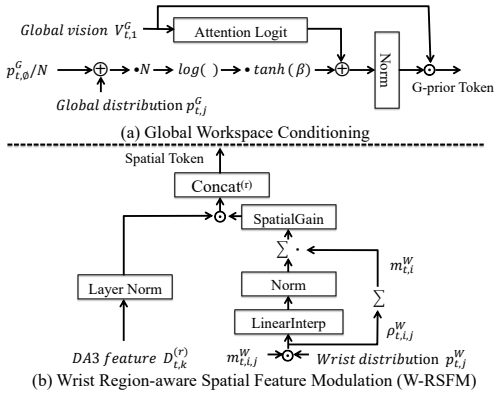

Figure 3: Detailed architecture of the proposed role-asymmetric spatial integration.

Building on these aligned query-wise maps, the *SpatialGain* module converts each normalized pattern into a log-density relative to the uniform distribution and standardizes it through spatial centering and RMS normalization. The standardized patterns are then aggregated using their retained responsibility masses. The aggregated response is scaled by $\alpha = 0.1$, exponentiated to obtain a positive multiplicative gain, and normalized by its spatial mean. This produces a unit-mean gain $g^W_{t,k}$ that reduces to identity modulation when no informative spatial preference is present, while softly emphasizing task-relevant regions without changing the overall feature scale.

The frozen DA3-BASE expert (Lin et al., 2025) extracts four levels of wrist-view features $\{\mathbf{D}^{(r)}_t\}_{r=1}^4$. After layer normalization and level-specific projection, W-RSFM applies the shared spatial gain and concatenates the modulated features into the Spatial Token sequence:

$$\mathbf{T}^{\mathrm{sp}}_t = \mathrm{Concat}^4_{r=1}\left(\left[g^W_{t,k}\,\Phi_r\left(\mathrm{LN}\left(\mathbf{D}^{(r)}_{t,k}\right)\right)\right]_{k=1}^{M}\right), \tag{4}$$

where Φ_r is the learned projection for feature level r, M is the number of geometry-grid positions, and $\mathbf{T}^{\mathrm{sp}}_t$ denotes the resulting Spatial Tokens. At the selected action-head blocks $\mathcal{L}_W = \{0, 6, 11, 12\}$, block-specific key and value projections incorporate $\mathbf{T}^{\mathrm{sp}}_t$ into the joint attention computation.

3.4 Two-Stage Training

Stage I: G/W route learning. With the pretrained VLM frozen, we optimize the G/W queries and their readout modules using the trajectory-derived targets from Section 3.2.2. For each view $v \in \{G, W\}$, the relevance target Y^v_t is converted into a distribution $\mathbf{y}^v_t$ over the visual patches and NULL. Valid patches receive their relevance values divided by the number of valid patches, invalid patches receive zero probability, and the residual mass is assigned to NULL. Given the predicted route $\mathbf{p}^v_t$, the routing objective is

$$\mathcal{L}_{\mathrm{route}} = -\frac{1}{|\mathcal{A}|}\sum_{(t,v)\in\mathcal{A}}\sum_{j\in\mathcal{P}_v\cup\{\emptyset\}} y^v_{t,j}\log p^v_{t,j} - \frac{\lambda_{\mathrm{MI}}}{|\mathcal{V}_{\mathcal{A}}|}\sum_{v\in\mathcal{V}_{\mathcal{A}}}\frac{\mathcal{I}_v}{\log K}, \tag{5}$$

Table 1: Comparison with competitive VLA baselines and geometry-enhanced methods on CALVIN ABC→D.

Method	Backbone Params.	Tasks completed in a row (%)					Avg. Len.
		1	2	3	4	5	
OpenVLA (Kim et al., 2024)	7B	91.3	77.8	62.0	52.1	43.5	3.27
UniVLA (Bu et al., 2025)	7B	95.5	85.8	75.4	66.9	56.5	3.80
π_0 (Black et al., 2026)	3B	93.8	85.0	76.7	68.6	60.1	3.84
VLA-Adapter (Wang et al., 2025)	0.5B	98.5	95.0	90.5	85.3	80.0	4.50
ReconVLA (Song et al., 2025)	7B	95.6	87.6	76.9	69.3	64.1	3.95
Spatial Forcing (Li et al., 2026b)	7B	93.6	85.8	78.4	72.0	64.6	3.94
FALCON (Zhang et al., 2026b)	1.6B	98.4	94.5	88.6	82.5	75.5	4.40
DreamVLA (Zhang et al., 2026a)	0.3B	98.2	94.6	89.5	83.4	78.1	4.44
ScopeVLA (Ours)	**0.5B**	**99.3**	**95.9**	**92.8**	**88.7**	**85.3**	**4.62**

where $\mathcal{A}$ contains the supervised observation–view pairs, $\mathcal{P}_v$ is the patch set for view v, and $\mathcal{V}_{\mathcal{A}}$ contains the corresponding views. The first term aligns the predicted routes with their spatial targets. The mutual information $\mathcal{I}_v$, computed from the target-weighted competitive assignments $\omega^v_{t,i,j}$ while excluding NULL, encourages query specialization and prevents query collapse. We set $K = 4$ and $\lambda_{\mathrm{MI}} = 0.02$.

Stage II: Joint policy training. The G/W queries and readout modules are initialized from Stage I, and the policy is trained while adapting the VLM through LoRA (Hu et al., 2021). The overall objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{act}} + \lambda \mathcal{L}_{\mathrm{route}}, \tag{6}$$

where $\mathcal{L}_{\mathrm{act}}$ is the standard ℓ_1 action-prediction loss and $\lambda = 0.02$. The two objectives update disjoint parameter sets: $\mathcal{L}_{\mathrm{route}}$ updates only the G/W queries and readout modules, whereas $\mathcal{L}_{\mathrm{act}}$ updates the trainable policy parameters, including the VLM LoRA parameters. The geometry expert remains frozen throughout both stages.

4 Experiments

4.1 Experimental Setup

Benchmarks. We evaluate ScopeVLA on CALVIN (Mees et al., 2022) and LIBERO-Plus (Fei et al., 2025). For CALVIN, we adopt the ABC→D protocol and report success rates for completing one to five consecutive subtasks and the average completed sequence length. For LIBERO-Plus, we train on the original LIBERO data (Liu et al., 2023) and evaluate zero-shot robustness across seven perturbation categories, reporting category-wise and overall success rates.

Implementation. ScopeVLA builds on a Prismatic VLM(Karamcheti et al., 2024) with a Qwen2.5-0.5B(Yang et al., 2024) language backbone and DINOv2/SigLIP(Oquab et al., 2023; Zhai et al., 2023) visual encoders, together with a VLA-Adapter-based action head. We use frozen DA3-BASE as the geometry expert and follow the two-stage training procedure described in Section 3.4. Training is conducted on NVIDIA GeForce RTX 5090 GPUs.

4.2 Simulation Results

Cross-environment manipulation on CALVIN. Table 1 compares performance under the ABC→D protocol. The cumulative success rates measure sustained execution over increasingly long instruction sequences, while the average sequence length summarizes overall performance. ScopeVLA achieves the best performance across all completion horizons, improving the average sequence length from 4.50 to 4.62 over the strongest baseline. Its advantage becomes more pronounced for longer sequences, with five-task success rising by 5.3 percentage points from 80.0% to 85.3%, indicating that task-conditioned spatial guidance helps preserve and exploit task-relevant spatial information throughout cross-environment sequential execution.

Table 2: Comparison with competitive VLA baselines and geometry-enhanced methods on LIBERO-Plus under zero-shot transfer from LIBERO.

Method	Backbone Params.	Perturbation categories							
		Camera	Robot	Language	Light	Background	Noise	Layout	Total
OpenVLA (Kim et al., 2024)	7B	0.8	3.5	23.0	8.1	34.8	15.2	28.5	15.6
UniVLA (Bu et al., 2025)	7B	1.8	**46.2**	69.6	69.0	81.0	21.2	31.9	43.9
π_0 (Black et al., 2026)	3B	13.8	6.0	58.8	85.0	81.4	**79.0**	68.9	53.6
VLA-Adapter (Wang et al., 2025)	0.5B	36.2	37.9	74.6	70.6	76.1	58.0	69.7	59.1
Spatial Forcing (Li et al., 2026b)	7B	20.1	13.4	40.9	29.1	33.4	25.7	39.3	29.1
DreamVLA (Zhang et al., 2026a)	0.3B	26.2	17.6	67.0	77.5	71.5	53.6	43.5	48.9
ScopeVLA (Ours)	0.5B	**55.3**	36.9	**87.3**	**91.2**	**87.9**	50.3	**75.6**	**67.4**

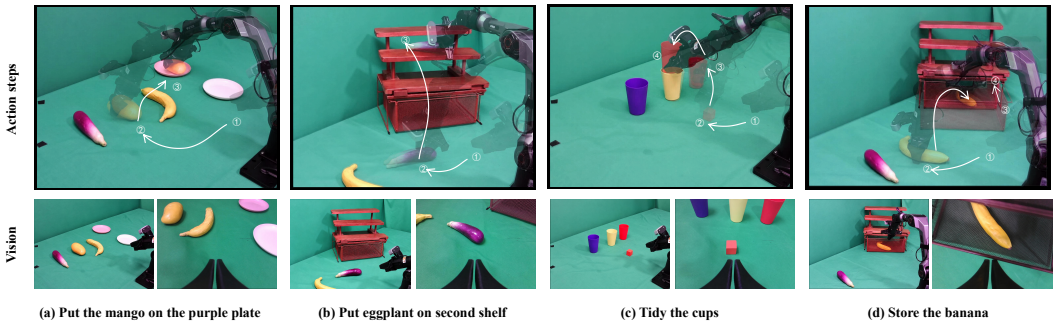

Figure 4: Real-world experiment setting and results.

Zero-shot robustness on LIBERO-Plus. Table 2 evaluates zero-shot transfer from LIBERO across seven perturbation categories. Among the evaluated parameter-comparable baselines, ScopeVLA achieves the highest overall score of 67.4, and ranks first in five of the seven categories. The particularly strong gains under camera, language, and layout perturbations indicate that task-conditioned, view-specialized spatial conditioning supports robust grounding across changes in viewpoint, instruction, and scene configuration.

4.3 Real-World Evaluation

Experimental setup and task design. We conduct real-world experiments on an AgileX robotic platform equipped with a 6-DoF single arm and a gripper, and use an Intel RealSense D435 as the depth camera. As shown in Figure 4, the evaluation comprises four tasks. To evaluate the ability of our method to identify manipulation targets and perceive task-relevant workspace regions, each task introduces a small number of distractor objects. The four tasks provide a structured evaluation of task-relevant workspace perception under visual distraction, encompassing spatially separated object–target associations, constrained or partially occluded placement regions, and sustained spatial reasoning throughout sequential manipulation. For each task, we collect 50 demonstration trajectories for training and conduct 20 independent evaluation trials for each method. A trial is counted as successful only when the specified terminal state is reached without human intervention.

Table 3: Success rates (%) on the real-world manipulation tasks.

Method	Mango–Plate	Eggplant–Shelf	Tidy Cups	Store Banana	Avg.
VLA-Adapter(Baseline)	80	75	35	55	61.25
ScopeVLA (Ours)	**95**	**85**	**60**	**80**	**80**

Experimental analysis. As shown in Table 3, ScopeVLA improves performance across all four tasks, increasing the average success rate from 61.25% to 80.0%. The gains are more pronounced on the challenging tasks (c) and (d), where success rates increase by 25 percentage points, from 35% to 60% and from 55% to 80%, respectively. These results provide empirical evidence that task-

conditioned spatial grounding and role-asymmetric spatial integration improve the policy's ability to localize task-relevant workspace regions and translate the resulting spatial cues into reliable action predictions under challenging manipulation conditions.

4.4 Ablation Studies

All variants are trained on the original LIBERO Object demonstrations and evaluated on LIBERO-Plus Object under the same protocol. Unless otherwise specified, we keep the policy backbone, training schedule, and evaluation procedure unchanged, and report the mean success rate over all perturbation settings.

Table 4: Ablation studies on LIBERO-Plus Object, including component design, view-specific geometry allocation, and geometry injection depth.

(a) Component design

Method	G-Condition	W-Geometry	W-RSFM	SR (%) ↑
A(Base)	×	×	×	46.3
B	✓	×	×	51.1
C	×	✓	×	56.5
D	✓	✓	×	61.2
E	×	✓	✓	59.7
F(Ours)	✓	✓	✓	**66.9**

(b) View-specific geometry allocation

Method	Geometry Allocation	SR (%) ↑
Base + W-RSFM	Wrist	**59.7**
Base + G-RSFM	Global	50.6
Base + G/W-RSFM	Global + Wrist	59.5

(c) Geometry injection depth

Method	Injected Blocks	SR (%) ↑
Uniform	$\{0, 8, 16, 23\}$	60.4
Early	$\{0, 4, 8, 11\}$	63.7
Late	$\{12, 16, 20, 23\}$	58.8
Ours	$\{0, 6, 11, 12\}$	**66.9**

Component ablation. We isolate the contributions of Global workspace conditioning (*G-Condition*), dense wrist-view geometry (*W-Geometry*), and wrist-route-guided spatial modulation (*W-RSFM*). As reported in Table 4(a), G-Condition and W-Geometry each improve the base policy, and combining them brings a further gain, indicating that scene-level task guidance and local geometric evidence provide complementary benefits. Applying W-RSFM on top of wrist geometry improves performance both with and without G-Condition, showing that the utility of dense geometry depends not only on its availability, but also on its task-conditioned allocation. Together, these components increase the success rate from 46.3% to 66.9%, yielding a 20.6-percentage-point improvement over the base policy.

View-specific geometry allocation. To determine where dense geometry is most effective, we apply the same route-guided modulation to the Global view, the Wrist view, or both. As shown in Table 4(b), allocating geometry to the Wrist view performs best; Global-view geometry is less effective, and combining both views yields no additional benefit. These results support ScopeVLA's role-asymmetric design: the Global view supplies scene-level task context, whereas the Wrist view provides interaction-relevant local geometry; uniformly applying dense geometry across views may instead introduce redundant or irrelevant spatial evidence.

Geometry injection depth. We compare different placements of geometric conditioning. As shown in Table 4(c), late-layer injection is least effective, while moving the interaction toward earlier layers yields consistent improvements. Our selected placement at $\{0, 6, 11, 12\}$ achieves the highest success rate of 66.9%, outperforming the uniform, early, and late alternatives. This result supports selecting the injection depths according to the layerwise VLM analysis and the characteristics of the VLA-Adapter-based action head, allowing geometric conditioning to interact with object-responsive and task-sensitive representations at appropriate stages.

5 Conclusion

We presented ScopeVLA, a task-grounded framework that selectively integrates spatial evidence according to the complementary roles of third-person and wrist views. By learning trajectory-supervised Global and Wrist routes from intermediate VLM representations, ScopeVLA uses global workspace priors to guide third-person visual attention and wrist responsibilities to modulate local geometric features. Controlled analyses, ablations, and evaluations in simulation and the real world consistently validate the effectiveness of task-grounded, view-specialized spatial conditioning for robust robotic manipulation.

AI USE STATEMENT

In this work, we used generative AI tools to assist with language polishing and improving the readability of the manuscript, as well as for literature retrieval and discovery, including identifying potentially relevant related work. We did not use generative AI tools for research ideation or execution, designing the methodology or experiments, implementing the proposed method, generating synthetic datasets, proving mathematical claims, or interpreting experimental results. We also did not use generative AI tools to draft substantive sections of the paper.

All AI-assisted writing was reviewed and revised by the authors. References identified with the assistance of generative AI tools were manually verified against the corresponding original papers and publication records before being included in the manuscript. We take responsibility for the final content of this work, including all text, claims, and references produced or identified with the aid of generative AI.

REFERENCES

Jinyue Bian, Zhaoxing Zhang, Zhengyu Liang, Shiwei Zheng, Shengtao Zhang, Rong Shen, Chen Yang, and Anzhou Hou. Vla-lpaf: Lightweight perspective-adaptive fusion for vision-language-action to enable more unconstrained robotic manipulation. *arXiv preprint arXiv:2509.18183*, 2025.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael 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, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2026. URL https://arxiv.org/abs/2410.24164.

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

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

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

Shizhe Chen, Paul Pacaud, and Cordelia Schmid. Pointact: Vision-language-action models with multi-scale point-action interaction. *arXiv preprint arXiv:2605.21414*, 2026a.

Yizhi Chen, Zhanxiang Cao, Xinyi Peng, Yixiao Zheng, Xiaxi Si, Yiheng Li, Liyun Yan, Keqi Zhu, Xueyun Chen, Shengcheng Fu, et al. Geoalign: Beyond semantics with state-guided spatial alignment in vla models. *arXiv preprint arXiv:2606.03240*, 2026b.

Qingyu Fan, Zhaoxiang Li, Jinrui Hu, Yi Lu, Wang Chen, Qiu Shen, Xiao-xiao Long, Yinghao Cai, Tao Lu, Shuo Wang, et al. Peafowl: Perception-enhanced multi-view vision-language-action for bimanual manipulation. *IEEE Robotics and Automation Letters*, 2026.

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

Minghao Guo, Meng Cao, Jiachen Tao, Rongtao Xu, Yan Yan, Xiaodan Liang, Ivan Laptev, and Xiaojun Chang. Glad: Geometric latent distillation for vision-language-action models. *arXiv preprint arXiv:2512.09619*, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Rishabh Jangir, Nicklas Hansen, Sambaran Ghosal, Mohit Jain, and Xiaolong Wang. Look closer: Bridging egocentric and third-person views with transformers for robotic manipulation. *IEEE Robotics and Automation Letters*, 7(2):3046–3053, 2022.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic vlms: Investigating the design space of visually-conditioned language models. *arXiv preprint arXiv:2402.07865*, 2024.

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

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

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.

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*, volume 2026, pp. 132324–132345, 2026b.

Peiyan Li, Yixiang Chen, Hongtao Wu, Xiao Ma, Xiangnan Wu, Yan Huang, Liang Wang, Tao Kong, and Tieniu Tan. Bridgevla: Input-output alignment for efficient 3d manipulation learning with vision-language models. *Advances in Neural Information Processing Systems*, 38:63635–63673, 2026c.

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

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

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

Mohan Liu, Zhihao Gu, Xuanyu Chen, Haitian Zhang, Kaimin Mao, Yan Wu, Wei-Yun Yau, and Lin Wang. Vistavla: Geometry-and semantic-aware 3d gaussian-grounded vla for robotic manipulation. *arXiv preprint arXiv:2607.12356*, 2026.

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

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Yuhao Pan, Haosong Peng, Zhengshen Zhang, Zhengyang Yan, Yalun Dai, Fushuo Huo, Chujie Wang, Tianyu Qi, Xiucheng Wang, Nan Cheng, et al. World-to-wrist: Task-conditioned future wrist modeling for fine-grained robot manipulation. *arXiv preprint arXiv:2608.05369*, 2026.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13529–13539, 2026.

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

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, et al. Smolvla: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025.

Wenxuan Song, Ziyang Zhou, Han Zhao, Jiayi Chen, Pengxiang Ding, Haodong Yan, Yuxin Huang, Feilong Tang, Donglin Wang, and Haoang Li. Reconvla: Reconstructive vision-language-action model as effective robot perceiver, 2025. URL https://arxiv.org/abs/2508.10333.

Guoheng Sun, Tingting Du, Kaixi Feng, Chenxiang Luo, Xingguo Ding, Zheyu Shen, Ziyao Wang, Yexiao He, and Ang Li. Rocket: Residual-oriented multi-layer alignment for spatially-aware vision-language-action models. *arXiv preprint arXiv:2602.17951*, 2026.

Lin Sun, Bin Xie, Yingfei Liu, Hao Shi, Tiancai Wang, and Jiale Cao. Geovla: Empowering 3d representations in vision-language-action models. *arXiv preprint arXiv:2508.09071*, 2025.

Hao Wang, Xiaobao Wei, Jingyang He, Chengyu Bai, Chun-Kai Fan, Jiajun Cao, Jintao Chen, Ying Li, Shanyu Rong, Ming Lu, et al. Vega: Visual encoder grounding alignment for spatially-aware vision-language-action models. *arXiv preprint arXiv:2605.10485*, 2026.

Yihao Wang, Pengxiang Ding, Lingxiao Li, Can Cui, Zirui Ge, Xinyang Tong, Wenxuan Song, Han Zhao, Wei Zhao, Pengxu Hou, Siteng Huang, Yifan Tang, Wenhui Wang, Ru Zhang, Jianyi Liu, and Donglin Wang. Vla-adapter: An effective paradigm for tiny-scale vision-language-action model, 2025. URL https://arxiv.org/abs/2509.09372.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report, 2024. URL https://arxiv.org/abs/2407.10671.

Bin Yu, Shijie Lian, Xiaopeng Lin, Zhaolong Shen, Yuliang Wei, Haishan Liu, Changti Wu, Hang Yuan, Bailing Wang, Cong Huang, et al. 3d-mix for vla: A plug-and-play module for integrating vggt-based 3d information into vision-language-action models. *arXiv preprint arXiv:2603.24393*, 2026.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, 2023.

Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, Xinqiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, He Wang, Zhizheng Zhang, et al. Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. *Advances in Neural Information Processing Systems*, 38: 24195–24228, 2026a.

Zhengshen Zhang, Hao Li, Yalun Dai, Zhengbang Zhu, Lei Zhou, Chenchen Liu, Dong Wang, Francis E. H. Tay, Sijin Chen, Ziwei Liu, Yuxiao Liu, Xinghang Li, and Pan Zhou. From spatial to actions: Grounding vision-language-action model in spatial foundation priors, 2026b. URL https://arxiv.org/abs/2510.17439.

Yanping Zhao, Hang Yu, Yiwei Wang, Chen Ye, Siyu Tian, Di Zhang, Qingjun Wang, Qian Chen, Junqiao Zhao, and Guang Chen. Gwm-vla: Geometry-aware latent world modeling for vision-language-action learning. *arXiv preprint arXiv:2608.07619*, 2026.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

A G/W Route Responses to Instruction and Scene Changes

Experimental design. We investigate whether the learned G/W routes are jointly conditioned on task instructions and current visual observations. We evaluate the Stage-I readout trained on CALVIN ABC using 96 CALVIN-D validation episodes spanning 32 tasks, with four frames sampled from each episode. As shown in Figure 5, we compare four conditions for both the Global and Wrist views. The correct-instruction condition evaluates whether the predicted routes align with the trajectory-derived targets. For the paraphrase and wrong-task conditions, the visual observations and targets are held fixed: the paraphrase preserves the original task semantics, whereas the wrong-task instruction refers to a disjoint task target. For the scene-change condition, we instead keep the instruction fixed and pair observations from different episodes of the same task at matched trajectory stages. Each changed scene is evaluated against its own trajectory-derived target.

Results. Under the correct instruction, the predicted Global and Wrist routes consistently cover the corresponding task regions, closely matching their trajectory-derived targets. Replacing the instruction with a same-task paraphrase causes little visible change, indicating that the readout is stable to variations in linguistic expression. In contrast, a wrong-task instruction substantially shifts or disperses the predicted route despite the image remaining unchanged. This contrast shows that the learned distributions depend on task semantics rather than solely reflecting image saliency.

The scene-change comparison provides complementary evidence. When the same instruction is applied to a different scene, both routes move with the scene-specific interaction region: the Global route adjusts its emphasis over the task workspace, while the Wrist route follows the local gripper–object configuration. Quantitatively, under the correct instruction, the Global and Wrist routes achieve Pearson similarities of 0.797 and 0.778 to their trajectory-derived targets, with KL divergences of 0.052 and 0.102, respectively. The KL values remain nearly unchanged under same-task paraphrases (0.051 and 0.102), but rise to 0.119 and 0.146 under wrong-task instructions. For scene changes, the predicted route is matched to the correct scene target in 86.97% of Global pairs and 91.90% of Wrist pairs. Together, these comparisons show that G/W routes are jointly conditioned on what the robot is instructed to do and where the relevant interaction occurs in the current observation, rather than reproducing a fixed task template or a generic visual-saliency map.

B Asymmetric Token Visibility

Figure 6 details the attention mask used to separate spatial grounding from action prediction. The visibility of the G/W queries is aligned with their respective roles: Global queries access the third-person view to capture scene-level workspace context, whereas Wrist queries access the wrist view to localize gripper–object interactions and task-relevant local geometry. Both groups attend to the instruction, and Wrist queries additionally access Global queries so that local grounding can be informed by global task context. Queries within each group are mutually visible to support coordinated spatial readout. Restricting each group to its designated visual stream discourages cross-view shortcuts and promotes view-specialized grounding. The original visual, instruction, and action streams retain their causal visibility, while action queries cannot attend directly to the G/W queries. Consequently, route-supervised representations remain isolated from the causal action stream and influence the policy only through Global Workspace Conditioning and W-RSFM.

C Trajectory-Grounded Workspace Targets

We construct the grounding targets from the complete end-effector trajectory and calibrated depth observations. Let $\{\mathbf{p}_i\}_{i=0}^{L-1}$ denote the trajectory and $k^\star$ an interaction anchor estimated from changes in end-effector position, orientation, and gripper aperture. Using the accumulated metric motion s_i, we define

$$q_i = \mathrm{clip}\left(\frac{s_i}{s_{k^\star}}, 0, 1\right), \quad P_i = 3q_i^2 - 2q_i^3, \quad g_i = g_0 + (1-g_0)P_i, \quad \sigma_i = \sigma_0 + (\sigma_1 - \sigma_0)P_i, \tag{7}$$

where P_i is fixed to one after the interaction anchor. We use $g_0 = 0.22$, $\sigma_0 = 0.015\,\mathrm{m}$, and $\sigma_1 = 0.045\,\mathrm{m}$. This schedule assigns weaker, narrower support to the approach trajectory and stronger, broader support to the interaction region.

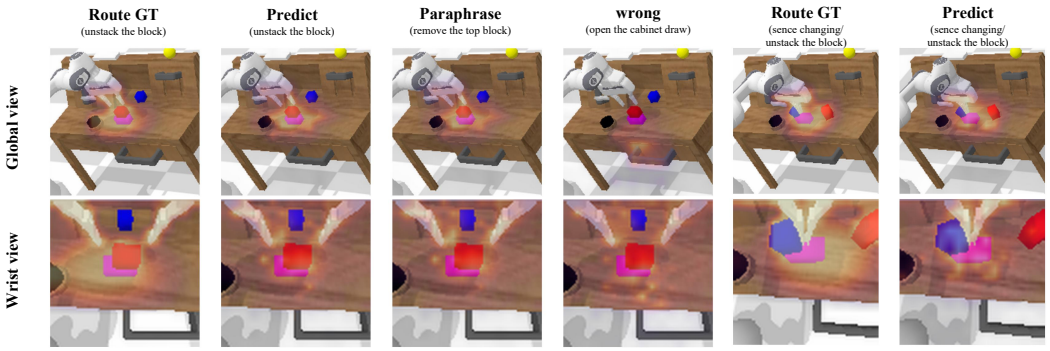

Figure 5: Visualization of task-relevant regions across different observations and viewpoints.

For a world point $\mathbf{x}$, let $d_i(\mathbf{x})$ be its distance to trajectory segment $\mathbf{p}_i \to \mathbf{p}_{i+1}$, and let $\bar{g}_i(\mathbf{x})$ and $\bar{\sigma}_i(\mathbf{x})$ denote the gain and bandwidth interpolated along that segment. The continuous workspace field is

$$F_\tau(\mathbf{x}) = \max_i \bar{g}_i(\mathbf{x}) \exp\left[-\frac{1}{2}\left(\frac{[d_i(\mathbf{x}) - r]_+}{\bar{\sigma}_i(\mathbf{x})}\right)^2\right], \tag{8}$$

where $r = 0.055\,\mathrm{m}$ approximates the gripper envelope. Max aggregation prevents densely sampled trajectory segments from receiving disproportionately large responses. The resulting field is fixed for the episode and retains both the approach corridor and the interaction region.

For view $v \in \{G, W\}$, each valid depth pixel $\mathbf{u}$ is back-projected into the world frame and evaluated in the same workspace field:

$$\mathbf{x}^v_{t,\mathbf{u}} = \mathrm{BackProject}\left(D^v_t(\mathbf{u}), \mathbf{K}^v_t, \mathbf{T}^v_{W\leftarrow C,t}\right), \qquad R^v_t(\mathbf{u}) = F_\tau(\mathbf{x}^v_{t,\mathbf{u}}). \tag{9}$$

Thus, the Global and Wrist views share a task-level spatial definition but receive different targets according to the surfaces visible from each camera.

The pixel responses are finally aggregated onto the VLM patch grid. For the valid pixels $\mathcal{U}^v_{t,j}$ in patch j, we average the highest $\eta = 20\%$ responses:

$$Y^v_{t,j} = \frac{1}{\kappa^v_{t,j}} \sum_{\mathbf{u}\in \mathrm{Top}_{\kappa^v_{t,j}}(\mathcal{U}^v_{t,j})} R^v_t(\mathbf{u}), \qquad \kappa^v_{t,j} = \left\lceil \eta |\mathcal{U}^v_{t,j}| \right\rceil. \tag{10}$$

This robust aggregation preserves small task-relevant structures that would be diluted by mean pooling, while reducing the sensitivity of max pooling to isolated depth or calibration noise. It produces the view-specific soft targets Y^G_t and Y^W_t. During route supervision, their visible relevance is assigned to the corresponding patches, while the residual probability is assigned to NULL. Depth, calibration, and trajectory information are used only to construct training targets and are not required at inference.

D SpatialGain in W-RSFM

Given the aligned and spatially normalized query-wise maps $\bar{\rho}^W_{t,i,k}$, SpatialGain first expresses each map as a log-density relative to the uniform distribution and standardizes it across the geometry grid:

$$\ell^W_{t,i,k} = \log\left(\max\left(M\bar{\rho}^W_{t,i,k}, \epsilon\right)\right), \qquad \widetilde{\ell}^W_{t,i,k} = \frac{\ell^W_{t,i,k} - \mu_{t,i}}{\max(\sigma_{t,i}, \epsilon)}, \tag{11}$$

where $\mu_{t,i}$ and $\sigma_{t,i}$ are the spatial mean and RMS deviation of $\ell^W_{t,i,k}$. The standardized patterns are then aggregated using their responsibility masses and converted into a positive, unit-mean gain:

$$s^W_{t,k} = \sum_{i=1}^{K} m^W_{t,i}\widetilde{\ell}^W_{t,i,k}, \qquad g^W_{t,k} = \frac{\exp(\alpha s^W_{t,k})}{\frac{1}{M}\sum_{k'=1}^{M}\exp(\alpha s^W_{t,k'})}, \tag{12}$$

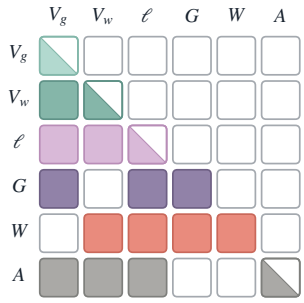

Figure 6: Asymmetric attention mask over the third-person visual (V_g), wrist-view visual (V_w), instruction (ℓ), Global-query (G), Wrist-query (W), and action-query (A) token groups. Rows denote queries and columns denote keys; filled, triangular, and empty cells indicate full, causal, and blocked visibility, respectively.

Table 5: Implementation hyperparameters of ScopeVLA.

Hyperparameter	**Value**
Global/Wrist queries	$4/4$
DA3 input resolution	504×504
DA3 decoder feature levels	4
Conditioning blocks	$\{0, 6, 11, 12\}$
Action prediction horizon	8
Optimizer	AdamW
LoRA rank	64
Per-device batch size (Stage I/II)	$32/8$
Gradient accumulation (Stage I/II)	$1/8$
Effective batch size (Stage I/II)	$32/64$
Learning rate (Stage I)	1×10^{-4}
Policy learning rate (Stage II)	2×10^{-4}
G/W and W-RSFM learning rate (Stage II)	1×10^{-4}
Learning-rate warm-up	first 10% of training

where $\alpha = 0.1$ controls the modulation strength and $\epsilon = 10^{-6}$ ensures numerical stability. This formulation emphasizes spatially relevant regions while maintaining $g^W_{t,k} > 0$ and $\frac{1}{M}\sum_k g^W_{t,k} = 1$. A uniform or uninformative route therefore yields identity modulation, preventing SpatialGain from changing the overall scale of the geometric features.

E Implementation Hyperparameters

The principal architectural and optimization hyperparameters are summarized in Table 5.

F Inference Efficiency

We measure inference efficiency on a single NVIDIA RTX 5090 with BF16 precision and batch size one. Each policy query takes two RGB views, the proprioceptive state, and a language instruction as input and predicts an eight-step action chunk. Both methods use the same fixed LIBERO Object observation and software stack, without quantization or model compilation. We perform 30 warm-up iterations and report the mean and standard deviation over 200 queries. End-to-end latency includes image preprocessing, tokenization, host-to-device transfer, policy inference, and action unnormalization, while excluding simulator rendering and environment stepping.

For a controlled architecture-level comparison, the base uses the same VLM, action-query, proprioceptive, and common action-head weights as ScopeVLA, while removing the G/W grounding, DA3

Table 6: Inference efficiency on an NVIDIA RTX 5090. Latency is measured per eight-step action chunk.

Method	**Params (B)**	**FLOPs (T)**	**Memory (GiB)**	**Latency (ms)**	**Query Rate (Hz)**
VLA-Adapter (Base)	1.471	1.496	3.54	31.48 ± 1.04	31.76
ScopeVLA (Ours)	1.629	1.799	4.13	52.67 ± 0.86	18.99

extraction, Global conditioning, and W-RSFM paths. All parameters of the plain VLA-Adapter action head are loaded from the common weights, with no randomly initialized components. As shown in Table 6, ScopeVLA requires 52.67 ms per policy query, compared with 31.48 ms for VLA-Adapter. The task-grounded spatial pathways therefore introduce 21.18 ms of latency, while increasing the counted computation from 1.496 T to 1.799 T FLOPs and peak allocated memory from 3.54 to 4.13 GiB. ScopeVLA retains a policy query rate of 18.99 Hz for an eight-step action chunk.