MMA: A SIMPLE AND FLEXIBLE MULTI-MODEL FRAMEWORK FOR ACTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models (VLMs) and vision-language-action models (VLAs) provide diverse pretrained representations for robot action learning, shaped by different data, architectures, and objectives. Combining their strengths is challenging because their interfaces differ, while running multiple models increases inference cost. We introduce MMA, a framework that jointly learns a fused policy and single-source branches that can be deployed independently. Learnable action queries and branch-local adaptation connect each source to a shared action interface and action head. An Action-Centric Mixture-of-Transformers (MoT) fuses branch representations, and a stop-gradient fused-query teacher transfers information to the branches through query alignment and relational matching. The resulting policies support either full-fusion or single-branch inference. On LIBERO, RoboTwin 2.0, and real-world manipulation tasks, the fused policy achieves average success rates of **99.2%**, **93.1%**, and **93.8%**, respectively, with absolute success-rate gains of **1.7%**, **5.5%**, and **15.0%** over the strongest matched single-source baselines. All three MMA branches also improve over their matched baselines across these evaluations. Each branch runs only its own source and the shared action head, providing a lower-cost deployment choice alongside fusion. MMA reuses pretrained components without access to their original pretraining data or additional large-scale pretraining. MMA offers a reference point for future work on combining pretrained VLMs and VLAs while balancing policy performance and deployment cost.

1 INTRODUCTION

Pretrained vision-language models (VLMs) connect visual observations with language, while vision-language-action models (VLAs) build on these representations to predict robot actions. Existing VLAs illustrate different approaches: RT-2 predicts action tokens, OpenVLA supports open fine-tuning, and π_0 generates continuous actions (Zitkovich et al., 2023; Kim et al., 2025b; Black et al., 2025b). Because these models differ in training data, architecture, and objectives, they may encode complementary information for control. This diversity motivates learning from multiple pretrained sources within one action policy.

Realizing this potential raises two coupled challenges. First, the sources expose features with different dimensions and token structures, so their information needs a common action interface. Second, using all sources at inference raises deployment cost. Thus, training should combine sources while preserving single-source deployment, and evaluation should assess both paths under their different inference inputs and costs.

Several lines of prior work address parts of these challenges. The Prismatic VLMs study compares vision encoders and language backbones, and Cambrian-1 aggregates features from multiple vision encoders (Karamcheti et al., 2024; Tong et al., 2024). At the interface level, BLIP-2 and VLA-Adapter use learnable queries to connect pretrained features to downstream modules, while Mixture-of-Transformers (MoT) separates parameter paths by token type (Li et al., 2023; Wang et al., 2026a; Liang et al., 2025). In parallel, teacher ensembles, Deep Mutual Learning, and ONE offer ways to share knowledge or jointly supervise models (Hinton et al., 2015; Zhang et al., 2018; Lan et al., 2018). Together, these ideas motivate fusion and transfer, which must be linked through action supervision to benefit independently deployed branches.

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.

To connect multi-source learning with single-source deployment, we introduce MMA. It supervises independent branches and a fused policy under a shared action head, while supporting either inference path. Our contributions are (1) a unified action interface that uses learnable queries, dimension mapping, and branch-local adaptation to connect heterogeneous pretrained sources to the shared head; (2) an Action-Centric MoT that fuses branch queries and contexts, together with stop-gradient query distillation from fusion to each deployable branch; and (3) matched evaluations on LIBERO, RoboTwin 2.0, and real-world tasks, where each MMA branch improves over its baseline and fusion leads our policies. MMA thus offers a reference point for future work on combining heterogeneous pretrained models under different robot deployment constraints.

2 Related Work

2.1 Vision-Language-Action Models

Generalist robot policies have been developed with large-scale datasets and adaptable architectures (Brohan et al., 2023; Open X-Embodiment Collaboration, 2024; Ghosh et al., 2024). Building on this foundation, VLAs connect vision-language representations to control through multimodal prompts, history, and action generation (Jiang et al., 2023; Li et al., 2024; Zitkovich et al., 2023; Kim et al., 2025b; Black et al., 2025b). Later work improves encoders, spatial features, action queries, and adaptation (Karamcheti et al., 2024; Tong et al., 2024; Qu et al., 2025; Kim et al., 2025a; Hou et al., 2025; Zheng et al., 2025; Black et al., 2025a; Zheng et al., 2026; Wang et al., 2026b). More specifically, Knowledge Insulating VLA isolates gradients between vision-language and action learning, while VLA-Adapter connects features through action queries and Bridge Attention (Driess et al., 2025; Wang et al., 2026a). MMA instead focuses on jointly reusing heterogeneous pretrained VLM and VLA sources through a shared action interface, training both a fused policy and independently deployable branches.

2.2 Multi-Model Collaboration and Knowledge Transfer

One line of work reuses pretrained knowledge through weight or skill merging, including Model Soups, Task Arithmetic, TIES-Merging, and MergeVLA (Wortsman et al., 2022; Ilharco et al., 2023; Yadav et al., 2023; Fu et al., 2026). Another connects sources through queries, encoder aggregation, or interacting parameter paths, as in BLIP-2, Cambrian-1, and MoT (Li et al., 2023; Tong et al., 2024; Liang et al., 2025). In robotics, TwinBrainVLA, RT-H, and Hi Robot connect semantic reasoning with embodied control (Yu et al., 2026; Belkhale et al., 2024; Shi et al., 2025).

Complementing these integration methods, distillation transfers teacher knowledge to a student (Hinton et al., 2015; Shen et al., 2019; Cao et al., 2025), while peer and online-ensemble methods provide mutual supervision (Zhang et al., 2018; Lan et al., 2018). Other methods preserve relationships between representations or distill action queries (Park et al., 2019; Tung & Mori, 2019; Wu et al., 2026). MMA links action-centric query fusion with fusion-to-branch distillation under a shared action head, jointly training a fused policy and branches that can be deployed alone.

3 Method

MMA learns a fused policy and independently deployable branches from multiple pretrained sources (Figure 1). A unified action interface connects each source to a shared action head; action-centric fusion combines their representations, and query distillation transfers fused information to the branches.

3.1 Problem Formulation

At time t, a language-conditioned policy receives image and instruction inputs $\mathbf{x}_t = (\mathbf{x}_t^{\text{img}}, \mathbf{x}_t^{\text{lang}})$ and an optional robot state $\mathbf{s}_t \in \mathbb{R}^{d_s}$. It predicts an action chunk of horizon H from demonstration samples:

$$\mathbf{A}_t = [\mathbf{a}_t; \ldots; \mathbf{a}_{t+H-1}] \in \mathbb{R}^{H \times d_a}, \qquad \mathcal{D} = \{(\mathbf{x}_n, \mathbf{s}_n, \mathbf{A}_n)\}_{n=1}^{N}. \tag{1}$$

d_s and d_a denote state and action dimensions, and N is the number of samples.

Let $k = 1, \ldots, K$ index pretrained sources. Each source accepts image–text and learnable query inputs and exposes hidden representations; hidden dimensions, context lengths, and backbones may differ. The unified interface maps their outputs to a shared action head.

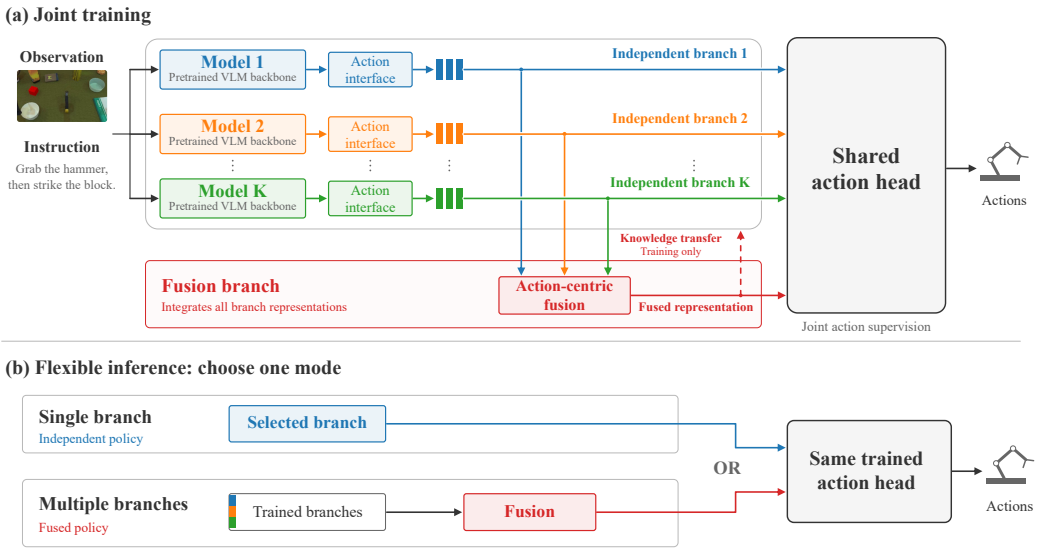


Figure 1: MMA training and inference. Source-specific interfaces connect branches to a shared action head, while action-centric fusion combines their queries and contexts. Red dashed lines mark fusion-to-branch transfer during training; inference uses either one branch or the fusion path.

MMA learns branch policies $\pi_k(\mathbf{A} \mid \mathbf{x}, \mathbf{s}; \theta_k, \phi)$ and a fusion policy $\pi_{\mathrm{F}}(\mathbf{A} \mid \mathbf{x}, \mathbf{s}; \theta_1, \ldots, \theta_K, \psi, \phi)$. Here, θ_k, ψ, and ϕ denote branch, fusion, and shared action-head parameters. Joint training couples the paths through action supervision and knowledge transfer; each branch can run alone at inference.

3.2 Unified Action Representation Interface

MMA uses action queries to extract task-relevant features from each source and maps them to a common dimension (Figure 2).

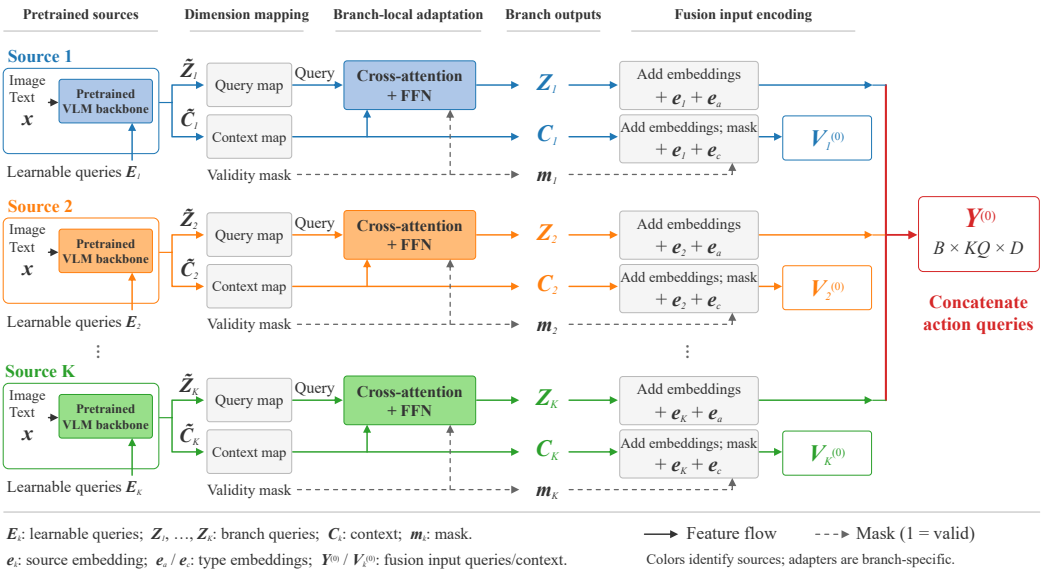


Figure 2: Unified action interface. Queries and contexts are mapped to a shared dimension; branch-local adaptation updates queries from their own context. Branch queries condition the shared action head, while queries and contexts enter fusion.

For batch size B, source k has hidden dimension d_k and learnable queries $\mathbf{E}_k \in \mathbb{R}^{Q \times d_k}$, where Q is the number of query slots, distinct from action horizon H. Feature extractor f_k processes the inputs and queries; we split its output at the query positions:

$$(\widetilde{\mathbf{Z}}_k, \widetilde{\mathbf{C}}_k) = \mathrm{Split}\big(f_k(\mathbf{x}; \mathbf{E}_k)\big), \widetilde{\mathbf{Z}}_k \quad \in \mathbb{R}^{B \times Q \times d_k}, \qquad \widetilde{\mathbf{C}}_k \in \mathbb{R}^{B \times T_k \times d_k}. \tag{2}$$

$\widetilde{\mathbf{Z}}_k$ contains action-query features and $\widetilde{\mathbf{C}}_k$ the remaining context; T_k is its padded length. Mask $\mathbf{m}_k \in \{0,1\}^{B\times T_k}$ marks valid context positions.

Let D be the shared dimension. Separate mappings $\mathcal{P}_k^{\mathrm{a}}$ and $\mathcal{P}_k^{\mathrm{c}}$ align queries and contexts when needed; a branch-local adapter lets queries read their own context:

$$\mathbf{C}_k = \mathcal{P}_k^{\mathrm{c}}(\widetilde{\mathbf{C}}_k), \qquad \mathbf{Z}_k = \mathrm{Adapt}_k\Big(\mathcal{P}_k^{\mathrm{a}}(\widetilde{\mathbf{Z}}_k), \mathbf{C}_k; \mathbf{m}_k\Big). \tag{3}$$

Adapt_k uses masked cross-attention from action queries to their source context, followed by a feed-forward update (Vaswani et al., 2017). Each branch outputs $(\mathbf{Z}_k, \mathbf{C}_k, \mathbf{m}_k)$ with $\mathbf{Z}_k \in \mathbb{R}^{B\times Q\times D}$ and $\mathbf{C}_k \in \mathbb{R}^{B\times T_k\times D}$. The shared head predicts actions from $\mathbf{Z}_k$; fusion uses all three outputs (**Appendix Table A.2**).

3.3 Action-Centric Multi-Model Fusion

Following Mixture-of-Transformers (MoT) (Liang et al., 2025), MMA separates action-query and context-token updates into Action and Context paths. Each path shares parameters across sources, while source embeddings preserve identity. Figure 3 shows information exchange and query pooling.

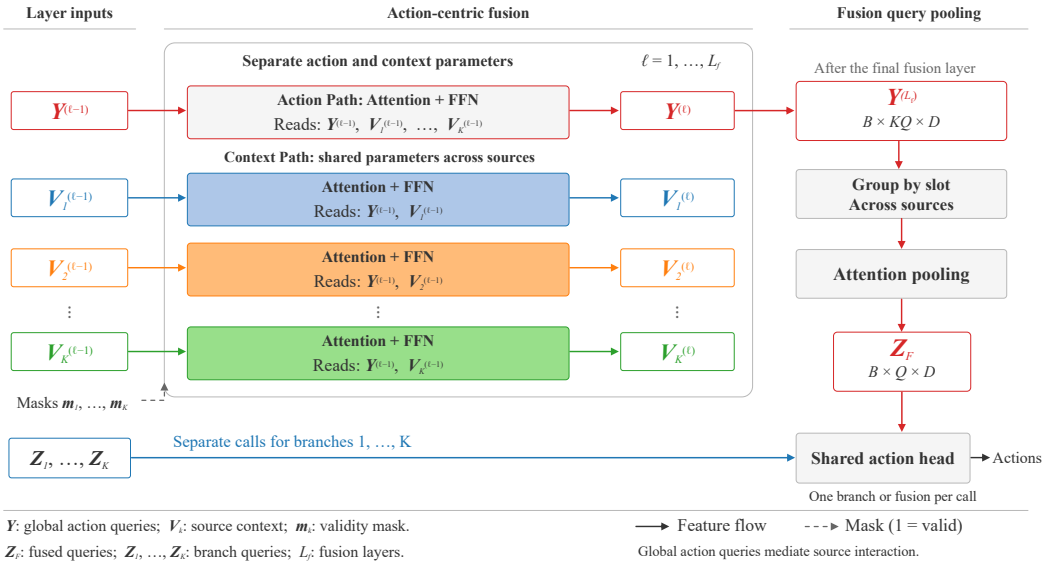


Figure 3: Action-centric fusion. Global action queries read all sources; each source context reads global queries and its own context. Action and Context paths use separate parameters, shared across sources within each path. Same-slot pooling produces fused action queries.

Input organization. Let $\mathbf{e}_k \in \mathbb{R}^D$ be a source embedding and $\mathbf{e}_{\mathrm{a}}, \mathbf{e}_{\mathrm{c}} \in \mathbb{R}^D$ be embeddings for the two token types. We concatenate action queries from all branches into a global sequence and retain each context separately:

$$\mathbf{Y}^{(0)} = [\mathbf{Z}_k + \mathbf{e}_k + \mathbf{e}_{\mathrm{a}}]_{k=1}^{K}, \qquad \mathbf{V}_k^{(0)} = \mathbf{m}_k \odot (\mathbf{C}_k + \mathbf{e}_k + \mathbf{e}_{\mathrm{c}}). \tag{4}$$

Square brackets concatenate query sequences, giving $\mathbf{Y}^{(0)} \in \mathbb{R}^{B\times KQ\times D}$; each masked context $\mathbf{V}_k^{(0)} \in \mathbb{R}^{B\times T_k\times D}$ remains source-specific.

Action-centric information exchange. MMA organizes cross-source interaction through action queries while retaining within-source context modeling. At fusion layer ℓ, global action queries read all action queries and contexts, whereas each context reads only the global action queries and its own context:

$$\begin{aligned} \mathbf{Y}^{(\ell)} &= \mathcal{B}_\ell^{\mathrm{a}}\Big(\mathbf{Y}^{(\ell-1)}; [\mathbf{Y}^{(\ell-1)}; \mathbf{V}_1^{(\ell-1)}; \ldots; \mathbf{V}_K^{(\ell-1)}]\Big), \\ \mathbf{V}_k^{(\ell)} &= \mathcal{B}_\ell^{\mathrm{c}}\Big(\mathbf{V}_k^{(\ell-1)}; [\mathbf{Y}^{(\ell-1)}; \mathbf{V}_k^{(\ell-1)}]\Big), \qquad \ell = 1, \ldots, L_{\mathrm{f}}. \end{aligned} \tag{5}$$

L_{f} is the number of fusion layers. $\mathcal{B}_\ell^{\mathrm{a}}$ and $\mathcal{B}_\ell^{\mathrm{c}}$ are type-specific attention and feed-forward updates: the sequence before the semicolon reads the key–value sequences after it. Both paths use the preceding layer state and update synchronously. Context keys follow source-specific masks, so contexts interact across sources through action queries rather than directly.

Fused query pooling. Reshape the final action-query sequence to $B \times K \times Q \times D$. Let $\mathbf{y}_{bkq}$ denote sample b, source k, and slot q, and let $\mathbf{S}_{bq} = [\mathbf{y}_{b1q}; \ldots; \mathbf{y}_{bKq}] \in \mathbb{R}^{K \times D}$. MMA pools sources at each corresponding query slot:

$$\overline{\mathbf{y}}_{bq} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{y}_{bkq}, \qquad \mathbf{z}_{bq}^{\mathrm{F}} = \mathrm{Pool}(\overline{\mathbf{y}}_{bq}, \mathbf{S}_{bq}), \qquad \mathbf{Z}_{\mathrm{F}} = [\mathbf{z}_{bq}^{\mathrm{F}}]_{b,q} \in \mathbb{R}^{B \times Q \times D}. \tag{6}$$

Pool attends from the normalized source mean to the same-slot source vectors, then applies residual attention and feed-forward updates. This yields one fused query per slot. $\mathbf{Z}_{\mathrm{F}}$ conditions the same action head as each branch. Fusion action loss updates the fusion module and participating sources.

3.4 Fusion-to-Branch Knowledge Transfer

MMA uses fused queries as an online teacher for each branch (Figure 4). Distillation combines slotwise alignment with teacher-anchored relational matching.

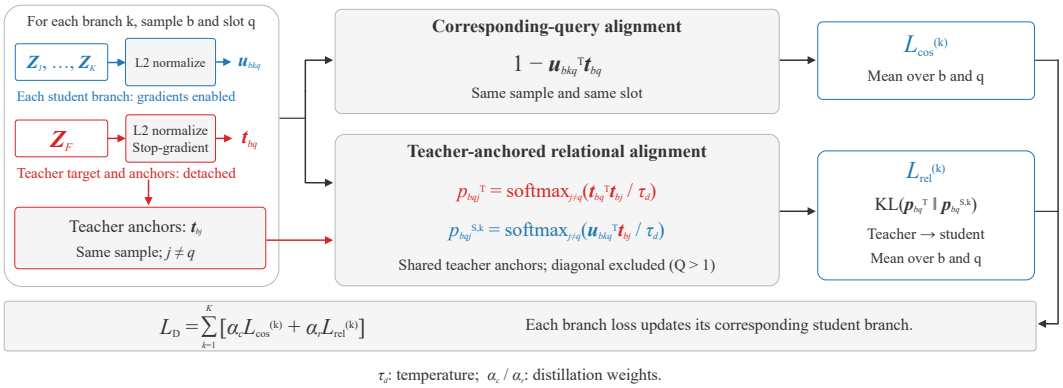


Figure 4: Fusion-to-branch distillation. Detached fused queries guide branch queries through corresponding-slot cosine alignment and teacher-anchored relational matching. Blue and red denote the student and teacher paths.

Corresponding-query alignment. Let ν denote feature-wise L2 normalization and sg stop-gradient. For sample b and slot q, define the detached teacher query and branch-k student query:

$$\mathbf{t}_{bq} = \mathrm{sg}\big(\nu(\mathbf{z}_{bq}^{\mathrm{F}})\big), \qquad \mathbf{u}_{bkq} = \nu\big(\mathbf{Z}_k[b, q, :]\big). \tag{7}$$

Corresponding teacher and student slots are aligned by cosine loss:

$$L_{\cos}^{(k)} = \frac{1}{BQ} \sum_{b=1}^{B} \sum_{q=1}^{Q} \big(1 - \mathbf{u}_{bkq}^{\top} \mathbf{t}_{bq}\big). \tag{8}$$

Teacher-anchored relational distillation. To preserve cross-query relationships, student and teacher queries are compared against the same teacher-query anchors within each sample. For temperature $\tau_{\mathrm{d}} > 0$ and $Q > 1$, define

$$p_{bqj}^{\mathrm{T}} = \mathrm{softmax}_{j \neq q}\big(\mathbf{t}_{bq}^{\top} \mathbf{t}_{bj} / \tau_{\mathrm{d}}\big), \quad p_{bqj}^{\mathrm{S},k} = \mathrm{softmax}_{j \neq q}\big(\mathbf{u}_{bkq}^{\top} \mathbf{t}_{bj} / \tau_{\mathrm{d}}\big). \tag{9}$$

Each softmax excludes the corresponding slot $j = q$ and normalizes over the other $Q - 1$ teacher anchors. The resulting distributions $\mathbf{p}_{bq}^{\mathrm{T}}$ and $\mathbf{p}_{bq}^{\mathrm{S},k}$ measure teacher and student relationships to the same anchors.

Teacher-to-student Kullback–Leibler (KL) divergence is combined with cosine alignment:

$$L_{\mathrm{rel}}^{(k)} = \frac{1}{BQ} \sum_{b=1}^{B} \sum_{q=1}^{Q} D_{\mathrm{KL}}\left(\mathbf{p}_{bq}^{\mathrm{T}} \,\|\, \mathbf{p}_{bq}^{\mathrm{S},k}\right), \quad L_{\mathrm{D}} = \sum_{k=1}^{K} \left(\alpha_{\mathrm{c}} L_{\cos}^{(k)} + \alpha_{\mathrm{r}} L_{\mathrm{rel}}^{(k)}\right). \tag{10}$$

$\alpha_{\mathrm{c}}, \alpha_{\mathrm{r}} \geq 0$ weight the two losses. Detached teacher queries make distillation update only the student branch; fusion action supervision updates the fusion teacher during joint training.

3.5 Joint Training and Inference

Joint training combines branch and fusion action losses with query distillation (Figure 5); parameter updates are summarized in **Appendix Table A.3**.

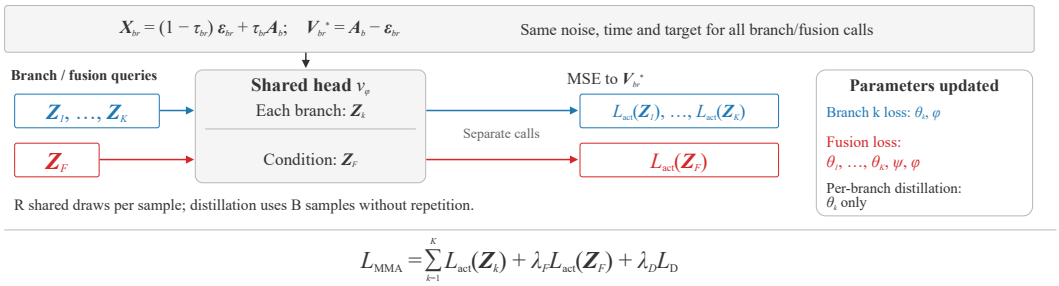


Figure 5: Joint action supervision. Branch and fusion queries condition the shared action head under common sampled noise, flow time, and velocity targets. Detached fused queries distill knowledge to branches; $r = 1, \ldots, R$ indexes repeated flow samples per example.

Shared action supervision. The shared action head is a Diffusion Transformer (DiT) velocity predictor v_ϕ conditioned on branch or fused queries and trained by flow matching (Lipman et al., 2023). π_0 also uses flow-matching action generation with pretrained vision-language features (Black et al., 2025b).

For target action $\mathbf{A}_b$, we sample Gaussian noise $\boldsymbol{\epsilon}_b$ and flow time τ_b, then construct $\mathbf{X}_b = (1 - \tau_b)\boldsymbol{\epsilon}_b + \tau_b \mathbf{A}_b$ and target velocity $\mathbf{V}_b^* = \mathbf{A}_b - \boldsymbol{\epsilon}_b$. Given a branch or fusion condition $\mathbf{Z}$, the action loss is

$$L_{\mathrm{act}}(\mathbf{Z}) = \frac{1}{BHd_a} \sum_{b=1}^{B} \mathbb{E}_{\boldsymbol{\epsilon}_b, \tau_b} \left[\| v_\phi(\mathbf{X}_b, c_b \mid \mathbf{Z}_b, \mathbf{s}_b) - \mathbf{V}_b^* \|_F^2 \right]. \tag{11}$$

c_b indexes flow time τ_b, and $\| \cdot \|_F$ is the Frobenius norm. All paths share sampled noise, flow time, and target velocity.

Joint optimization. MMA combines branch action supervision, fusion action supervision, and query distillation as

$$L_{\mathrm{MMA}} = \sum_{k=1}^{K} L_{\mathrm{act}}(\mathbf{Z}_k) + \lambda_{\mathrm{F}} L_{\mathrm{act}}(\mathbf{Z}_{\mathrm{F}}) + \lambda_{\mathrm{D}} L_{\mathrm{D}}, \qquad \lambda_{\mathrm{F}}, \lambda_{\mathrm{D}} \geq 0. \tag{12}$$

The three terms supervise branch actions, fused actions, and fusion-to-branch transfer, respectively. All action losses update the shared head; **Appendix Table A.3** gives their parameter-update scope.

Inference and deployment. At inference, the shared head predicts an action chunk from either $\mathbf{Z}_k$ or $\mathbf{Z}_{\mathrm{F}}$. Branch deployment runs one source; fusion deployment runs all sources and the fusion module. They trade compute for access to multi-source information.

4 Experiments and Analysis

We evaluate whether joint training improves the fusion policy and individual branches and which components matter. Same-source independently trained policies provide baselines.

Table 1: LIBERO success rates (%).

Method	Venue/year	Spatial	Object	Goal	Long	Avg. ↑
OpenVLA (Kim et al., 2025b)	*CoRL* '24	84.7	88.4	79.2	53.7	76.5
OpenVLA-OFT (Kim et al., 2025a)	*RSS* '25	97.6	98.4	97.9	94.5	97.1
$\pi_{0.5}$ (Black et al., 2025a)	*CoRL* '25	98.8	98.2	98.0	92.4	96.9
π_0 (Black et al., 2025b)	*RSS* '25	96.8	98.8	95.8	85.2	94.2
ACoT-VLA (Frozen) (Zhong et al., 2026)	*CVPR* '26	99.4	99.6	98.8	96.0	98.5
LingBot-VA (Li et al., 2026)	*RSS* '26	98.5	99.6	97.2	**98.5**	98.5
VLA-Adapter-Pro (Wang et al., 2026a)	*AAAI* '26	99.6	99.6	98.2	96.4	98.5
X-VLA (Zheng et al., 2026)	*ICLR* '26	98.2	98.6	97.8	97.6	98.1
MemoryVLA (Shi et al., 2026)	*ICLR* '26	98.4	98.4	96.4	93.4	96.7
Fast-WAM-Joint (Yuan et al., 2026)	*arXiv* '26	99.6	99.4	98.2	96.8	98.5
ABot-M0 (Yang et al., 2026b)	*arXiv* '26	98.8	**99.8**	99.0	96.6	98.6
MolmoAct2-Think (Fang et al., 2026)	*arXiv* '26	98.8	**99.8**	98.5	95.4	98.1
Qwen3.5: Baseline	—	96.0	97.0	97.6	93.8	96.1
MMA: Qwen3.5 branch	—	**99.8**	**99.8**	98.2	96.0	98.5+2.4%
Cosmos-Reason2: Baseline	—	97.0	97.4	97.2	94.2	96.5
MMA: Cosmos-Reason2 branch	—	99.6	98.2	98.6	96.4	98.2+1.7%
ABot-M0: Baseline	—	98.8	99.2	97.0	94.8	97.5
MMA: ABot-M0 branch	—	**99.8**	**99.8**	99.4	96.6	98.9+1.4%
MMA: Fusion policy	—	**99.8**	**99.8**	**99.6**	97.4	**99.2**

Published rows use their reported protocols; the four-suite π_0 result follows RIPT-VLA (Tan et al., 2025). Published Avg. values follow the cited sources. Small red values after MMA branch averages show same-model success-rate gains in %.

4.1 Experimental Setup

We measure closed-loop success on the four LIBERO suites (Liu et al., 2023), 50 RoboTwin 2.0 tasks in Clean and Randomized conditions (Chen et al., 2025), and four real-world tasks. MMA uses Qwen3.5-0.8B, Cosmos-Reason2-2B, and ABot-M0, with independently trained same-source policies as matched baselines. **Appendix A** details the experimental setup (**Section A.1**): benchmark training and model sizes (**Section A.1.1**, **Table A.1**), mechanism-analysis sampling (**Section A.1.2**), and the real-world protocol and task images (**Section A.1.4**, **Figure A.1**). ***Code is provided in the supplementary materials.***

4.2 Main Results

4.2.1 LIBERO

Table 1 reports the four LIBERO suites and their equally weighted average.

The fusion policy averages 99.2% on LIBERO, an absolute 1.7% gain over the strongest same-source baseline, ABot-M0 (Table 1). Each MMA branch improves over its own baseline by 1.4–2.4%, with the largest average gain on Qwen3.5. Fusion exceeds the strongest MMA branch by 0.3%. Long is the lowest-scoring suite for every MMA path; fusion scores 97.4% there, 0.8% above the best branch, while reaching at least 99.6% on the other three suites.

4.2.2 RoboTwin

The fusion policy averages 93.1% on RoboTwin, an absolute 5.5% gain over the strongest same-source baseline (Table 2). All three MMA branches improve over their matched baselines in both conditions; their average gains range from 4.5% to 7.7%. Fusion reaches 93.3% in Clean and 92.9% in Randomized conditions, exceeding the best branch by 0.8% and 0.9%, respectively. Its 0.4% Clean–Randomized gap is also smaller than that of the strongest same-source baseline (0.8%).

4.2.3 Real-World Experiments

The fusion policy averages 93.8% across the real-world tasks, an absolute 15.0% gain over the strongest same-source baseline and 8.8% over the strongest MMA branch (Table 3). All three branches improve over their matched baselines, with Cosmos-Reason2 showing the largest average gain (12.5%). Fusion leads the best branch by 5.0% on Placement and Replacement, 15.0% on Sorting, and 10.0% on Stacking. It succeeds in all 20 Sorting trials, while its other task success rates range from 90% to 95%.

Table 2: RoboTwin 2.0 success rates under Data Scaling Settings (%).

Method	Venue/year	Clean	Randomized	Avg. ↑
π_0 (Black et al., 2025b)	*RSS* '25	65.9	58.4	62.2
$\pi_{0.5}$ (Black et al., 2025a)	*CoRL* '25	82.7	76.8	79.8
LingBot-VA (Li et al., 2026)	*RSS* '26	92.9	91.6	92.2
Motus (Bi et al., 2026)	*CVPR* '26	88.7	87.0	87.8
HoloBrain-0-QW (Lin et al., 2026)	*arXiv* '26	91.9	92.3	92.1
Fast-WAM (Yuan et al., 2026)	*arXiv* '26	91.9	91.8	91.8
ABot-M0 (Yang et al., 2026b)	*arXiv* '26	86.1	85.1	85.6
LingBot-VLA (Wu et al., 2026)	*arXiv* '26	88.6	86.7	87.6
FlowWAM (Chen et al., 2026)	*arXiv* '26	92.9	92.1	92.5
InternVLA-A1 (Cai et al., 2026)	*arXiv* '26	89.4	89.6	89.5
VLAct (Yang et al., 2026a)	*arXiv* '26	92.5	90.8	91.7
Qwen3.5: Baseline	—	84.4	82.0	83.2
MMA: Qwen3.5 branch	—	90.8+6.4%	89.5+7.5%	90.2+7.0%
Cosmos-Reason2: Baseline	—	84.6	84.4	84.5
MMA: Cosmos-Reason2 branch	—	92.5+7.9%	91.8+7.4%	92.2+7.7%
ABot-M0: Baseline	—	88.0	87.2	87.6
MMA: ABot-M0 branch	—	92.2+4.2%	92.0+4.8%	92.1+4.5%
MMA: Fusion policy	—	**93.3**	**92.9**	**93.1**

Published rows follow reported Data Scaling protocols (50 Clean and 500 Randomized demonstrations per task); the π_0, $\pi_{0.5}$, InternVLA-A1, and VLAct values follow VLAct Table 2. Our Avg. is the mean of Clean and Randomized. Small red values show same-model success-rate gains in %.

Table 3: Real-world results (success rate, %).

Policy	Placement	Replacement	Sorting	Stacking	Avg.
RDT-1B (Liu et al., 2025)	75.0	55.0	70.0	50.0	62.5
X-VLA (Zheng et al., 2026)	85.0	65.0	80.0	60.0	72.5
VLA-Adapter (Wang et al., 2026a)	70.0	80.0	85.0	60.0	73.8
OpenVLA-OFT (Kim et al., 2025a)	75.0	85.0	90.0	65.0	78.8
π_0 (Black et al., 2025b)	90.0	70.0	85.0	75.0	80.0
$\pi_{0.5}$ (Black et al., 2025a)	90.0	75.0	80.0	85.0	82.5
Qwen3.5: Baseline	70.0	65.0	70.0	65.0	67.5
MMA: Qwen3.5 branch	85.0	75.0	80.0	70.0	77.5+10.0%
Cosmos-Reason2: Baseline	75.0	75.0	65.0	70.0	71.3
MMA: Cosmos-Reason2 branch	90.0	85.0	85.0	75.0	83.8+12.5%
ABot-M0: Baseline	80.0	80.0	85.0	70.0	78.8
MMA: ABot-M0 branch	90.0	85.0	85.0	80.0	85.0+6.2%
MMA: Fusion policy	**95.0**	**90.0**	**100.0**	**90.0**	**93.8**

Each task has 20 trials. Small red values show same-model success-rate gains in %.

4.3 Ablation Studies

Table 4 compares all four MMA paths with variants that remove query distillation, replace learned fusion with query averaging, or remove relational distillation. Exact settings are in **Appendix A.1.1**.

Full MMA has the highest average success among these settings for both fusion and every branch on both benchmarks. Replacing learned fusion with query averaging gives the largest RoboTwin fusion drop, from 93.1% to 88.3%.

4.4 Mechanism Evidence

Appendix B presents the mechanism analyses. Module changes and action readout are examined in **Section B.1** (**Figures B.1 and B.2**); query-slot covariance is examined in **Section B.2** (**Figure B.3**). In the Qwen–Cosmos–ABot configuration, source reading changes during training, and source interventions alter fused queries and action predictions (**Section B.3**, **Figures B.4 and B.5**). Teacher–branch errors, relational matching, and paired changes in query alignment and flow error are examined in **Section B.4** (**Figures B.6, B.7, and B.8**).

Table 4: Ablation of key MMA components (success rate, %). Green suffixes show success-rate decreases from full MMA in %.

Ablation setting	Policy branch	LIBERO					RoboTwin 2.0		
		Spatial	Object	Goal	Long	Avg. ↑	Clean	Rand.	Avg. ↑
Full MMA	Qwen3.5	99.8	99.8	98.2	96.0	98.5	90.8	89.5	90.2
	Cosmos-Reason2	99.6	98.2	98.6	96.4	98.2	92.5	91.8	92.2
	ABot-M0	99.8	99.8	99.4	96.6	98.9	92.2	92.0	92.1
	Fusion branch	99.8	99.8	99.6	97.4	99.2	93.3	92.9	93.1
No query distillation ($\lambda_D = 0$)	Qwen3.5	98.6	98.4	98.0	94.6	97.4-1.1%	86.8-4.0%	85.2-4.3%	86.0-4.2%
	Cosmos-Reason2	98.2	98.0	97.8	95.0	97.3-0.9%	87.6-4.9%	86.9-4.9%	87.3-4.9%
	ABot-M0	99.6	99.6	98.2	95.8	98.3-0.6%	88.7-3.5%	87.9-4.1%	88.3-3.8%
	Fusion branch	99.6	99.6	98.6	95.8	98.4-0.8%	90.1-3.2%	89.3-3.6%	89.7-3.4%
Mean query fusion	Qwen3.5	98.2	98.0	98.0	94.6	97.2-1.3%	86.1-4.7%	84.3-5.2%	85.2-5.0%
	Cosmos-Reason2	97.8	98.0	97.8	95.0	97.2-1.0%	86.5-6.0%	86.1-5.7%	86.3-5.9%
	ABot-M0	99.6	99.6	97.8	95.6	98.2-0.7%	88.6-3.6%	87.8-4.2%	88.2-3.9%
	Fusion branch	99.6	99.6	98.4	95.6	98.3-0.9%	88.7-4.6%	87.9-5.0%	88.3-4.8%
No relational distillation ($\alpha_r = 0$)	Qwen3.5	98.6	98.6	98.0	94.8	97.5-1.0%	88.5-2.3%	87.2-2.3%	87.9-2.3%
	Cosmos-Reason2	98.4	98.0	97.8	95.4	97.4-0.8%	90.2-2.3%	89.5-2.3%	89.9-2.3%
	ABot-M0	99.6	99.6	98.6	96.0	98.5-0.4%	90.1-2.1%	89.6-2.4%	89.9-2.2%
	Fusion branch	99.6	99.6	98.6	96.2	98.5-0.7%	91.1-2.2%	90.6-2.3%	90.9-2.2%

Inference cost. Table 5 and Figure 6 report inference costs for Qwen3.5-0.8B, Cosmos-Reason2-2B, and ABot-M0. The fusion path takes 136.50 ms, 9.76 GiB of peak GPU memory, and 5.29 B resident parameters; the single-branch latencies range from 38.40 to 84.72 ms. **Appendix A.1.3** gives the measurement scope.

Table 5: Measured inference latency, memory, and parameter count.

Policy	Latency (ms)	Memory (GiB)	Parameters (B)
Qwen3.5-0.8B	38.40	2.18	0.90
Cosmos-Reason2-2B	59.58	3.69	1.73
ABot-M0	84.72	5.54	3.63
MMA fusion	136.50	9.76	5.29

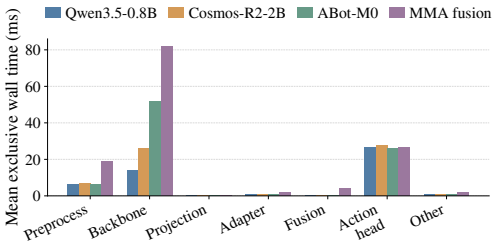


Figure 6: Mean exclusive wall time.

5 Limitations and Future Work

Limitations. Joint training processes every source, so compute and memory grow roughly linearly with the number of similarly sized models. Fusion and distillation add further cost. MMA also requires query inputs, accessible hidden representations, and valid context masks.

Future work. Parameter-efficient adaptation, selective branch updates, and lighter knowledge transfer could reduce training cost. Controlled studies across model numbers, architectures, and scales would clarify when collaboration helps and guide task-specific source selection.

6 Conclusion

This paper studies how to combine transferable knowledge from multiple pretrained sources for robot action learning and proposes the MMA multi-model framework. MMA uses learnable action queries, dimension mapping when needed, and branch-local context adaptation to build a unified interface. Under a shared action head, it jointly learns independent branches and a fused policy, and fusion-to-branch query distillation connects multi-source integration with independent policy learning. This design supports both full fusion inference and fixed single-branch deployment, allowing source reuse and deployment choices to be studied in the same training framework. MMA offers a path to multi-model collaboration for VLA construction, with source combinations that can be studied against task demands and deployment resources across robot settings.

AI USE STATEMENT

Generative AI tools assisted with English translation, language editing, figure preparation and terminology consistency, LaTeX formatting, and checks of numerical and cross-reference consistency. The authors are responsible for the final manuscript and its claims.

REFERENCES

Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quan Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. RT-H: Action hierarchies using language. In *Proceedings of Robotics: Science and Systems*, 2024. URL https://www.roboticsproceedings.org/rss20/p049.html.

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 *Proceedings of the 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 *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 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, et al. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems*, 2025b. doi: 10.15607/RSS.2025.XXI.010. URL https://www.roboticsproceedings.org/rss21/p010.html.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, et al. RT-1: Robotics transformer for real-world control at scale. In *Proceedings of Robotics: Science and Systems*, 2023. doi: 10.15607/RSS.2023.XIX.025. URL https://roboticsproceedings.org/rss19/p025.html.

Junhao Cai, Zetao Cai, Jiafei Cao, Yilun Chen, Zeyu He, Lei Jiang, Hang Li, Hengjie Li, Yang Li, Yufei Liu, Yanan Lu, Qi Lv, Haoxiang Ma, Jiangmiao Pang, Yu Qiao, Zherui Qiu, Yanqing Shen, Xu Shi, Yang Tian, Bolun Wang, Hanqing Wang, Jiaheng Wang, Tai Wang, Xueyuan Wei, Chao Wu, Yiman Xie, Boyang Xing, Yuqiang Yang, Yuyin Yang, Qiaojun Yu, Feng Yuan, Jia Zeng, Jingjing Zhang, Shenghan Zhang, Shi Zhang, Zhuoma Zhaxi, Bowen Zhou, Yuanzhen Zhou, Yunsong Zhou, Hongrui Zhu, Yangkun Zhu, and Yuchen Zhu. InternVLA-A1: Unifying Understanding, Generation and Action for Robotic Manipulation. *arXiv preprint arXiv:2601.02456*, 2026. URL https://arxiv.org/abs/2601.02456.

Jiajun Cao, Yuan Zhang, Tao Huang, Ming Lu, Qizhe Zhang, Ruichuan An, Ningning Ma, and Shanghang Zhang. MoVE-KD: Knowledge distillation for VLMs with mixture of visual encoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Cao_MoVE-KD_Knowledge_Distillation_for_VLMs_with_Mixture_of_Visual_Encoders_CVPR_2025_paper.html.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan-ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. RoboTwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025. URL https://arxiv.org/abs/2506.18088.

Yixiang Chen, Peiyan Li, Yuan Xu, Qisen Ma, Jiabing Yang, Kai Wang, Jianhua Yang, Dong An, He Guan, Gaoteng Liu, Jianlou Si, Jun Huang, Jing Liu, Nianfeng Liu, Yan Huang, and Liang Wang. FlowWAM: Optical Flow as a Unified Action Representation for World Action Models. *arXiv preprint arXiv:2607.13017*, 2026. URL https://arxiv.org/abs/2607.13017.

Danny Driess, Jost Tobias Springenberg, Brian Ichter, Lili Yu, Adrian Li-Bell, Karl Pertsch, Allen Z. Ren, Homer Walke, Quan Vuong, Lucy Xiaoyang Shi, and Sergey Levine. Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://papers.neurips.cc/paper_files/paper/2025/hash/94e936034d12bcd04834ec2773f02aff-Abstract-Conference.html.

Haoquan Fang, Jiafei Duan, Donovan Clay, Sam Wang, Shuo Liu, Weikai Huang, Xiang Fan, Wei-Chuan Tsai, Shirui Chen, Yi Ru Wang, Shanli Xing, Jaemin Cho, Jae Sung Park, Ainaz Eftekhar, Peter Sushko, Karen Farley, Angad Wadhwa, Cole Harrison, Winson Han, Ying-Chun Lee, Eli VanderBilt, Rose Hendrix, Suveen Ellawela, Lucas Ngoo, Joyce Chai, Zhongzheng Ren, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoAct2: Action reasoning models for real-world deployment. *arXiv preprint arXiv:2605.02881*, 2026. URL https://arxiv.org/abs/2605.02881.

Yuxia Fu, Zhizhen Zhang, Yuqi Zhang, Zijian Wang, Zi Huang, and Yadan Luo. MergeVLA: Cross-skill model merging toward a generalist vision-language-action agent. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026. URL https://arxiv.org/abs/2511.18810.

Dibya Ghosh, Homer Rich Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, et al. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.090. URL https://www.roboticsproceedings.org/rss20/p090.html.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. doi: 10.48550/arXiv.1503.02531. URL https://arxiv.org/abs/1503.02531.

Zhi Hou, Tianyi Zhang, Yuwen Xiong, Haonan Duan, Hengjun Pu, Ronglei Tong, Chengyang Zhao, Xizhou Zhu, Yu Qiao, Jifeng Dai, and Yuntao Chen. Dita: Scaling Diffusion Transformer for Generalist Vision-Language-Action Policy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7686–7697, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Hou_Dita_Scaling_Diffusion_Transformer_for_Generalist_Vision-Language-Action_Policy_ICCV_2025_paper.html.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. VIMA: Robot manipulation with multimodal prompts. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14975–15022. PMLR, 2023. URL https://proceedings.mlr.press/v202/jiang23b.html.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic VLMs: Investigating the design space of visually-conditioned language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 23123–23144. PMLR, 2024. URL https://proceedings.mlr.press/v235/karamcheti24a.html.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. In *Proceedings of 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, et al. OpenVLA: An open-source vision-language-action model. In *Proceedings of the 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2025b. URL https://proceedings.mlr.press/v270/kim25c.html.

Xu Lan, Xiatian Zhu, and Shaogang Gong. Knowledge distillation by on-the-fly native ensemble. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/94ef7214c4a90790186e255304f8fd1f-Abstract.html.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*. PMLR, 2023. URL https://proceedings.mlr.press/v202/li23q.html.

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 XXII*, 2026. doi: 10.

15607/RSS.2026.XXII.016. URL https://www.roboticsproceedings.org/rss22/p016.html.

Xinghang Li, Minghuan Liu, Hanbo Zhang, Cunjun Yu, Jie Xu, Hongtao Wu, Chilam Cheang, Ya Jing, Weinan Zhang, Huaping Liu, Hang Li, and Tao Kong. Vision-language foundation models as effective robot imitators. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/71639c317fb0bf398835627b4418693e-Abstract-Conference.html.

Weixin Liang, Lili Yu, Liang Luo, Srinivasan Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=Nu6N69i8SB.

Xuewu Lin, Tianwei Lin, Yun Du, Hongyu Xie, Yiwei Jin, Jiawei Li, Shijie Wu, Qingze Wang, Mengdi Li, Mengao Zhao, Ziang Li, Chaodong Huang, Hongzhe Bi, Lichao Huang, and Zhizhong Su. HoloBrain-0 Technical Report. *arXiv preprint arXiv:2602.12062*, 2026. URL https://arxiv.org/abs/2602.12062.

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, 2023. doi: 10.52202/075280-1939. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/8c3c666820ea055a77726d66fc7d447f-Abstract-Datasets_and_Benchmarks.html.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. RDT-1B: a diffusion foundation model for bimanual manipulation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/49f80e4d2471ad4f2edf4f5f1ab62339-Abstract-Conference.html.

NVIDIA. Cosmos-Reason2: Physical AI common sense and embodied reasoning models. Hugging Face model card for Cosmos-Reason2-2B, 2026. URL https://huggingface.co/nvidia/Cosmos-Reason2-2B.

Open X-Embodiment Collaboration. Open x-embodiment: Robotic learning datasets and RT-X models. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2024. URL https://arxiv.org/abs/2310.08864.

Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3967–3976, 2019. doi: 10.1109/CVPR.2019.00409. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Park_Relational_Knowledge_Distillation_CVPR_2019_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 *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.011. URL https://www.roboticsproceedings.org/rss21/p011.html.

Qwen Team. Qwen3.5-0.8B. Hugging Face model card, 2026. URL https://huggingface.co/Qwen/Qwen3.5-0.8B.

Chengchao Shen, Mengqi Xue, Xinchao Wang, Jie Song, Li Sun, and Mingli Song. Customizing student networks from heterogeneous teachers via adaptive knowledge amalgamation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3503–3512, 2019. doi: 10.1109/ICCV.2019.00360. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Shen_Customizing_Student_Networks_From_Heterogeneous_Teachers_via_Adaptive_Knowledge_Amalgamation_ICCV_2019_paper.html.

Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. MemoryVLA: Perceptual-Cognitive Memory in Vision-Language-Action Models for Robotic Manipulation. In *International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10011504.

Lucy Xiaoyang Shi, Brian Ichter, Michael Robert Equi, Liyiming Ke, Karl Pertsch, Quan Vuong, James Tanner, Anna Walling, Haohuan Wang, Niccolo Fusai, Adrian Li-Bell, Danny Driess, Lachy Groom, Sergey Levine, and Chelsea Finn. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp.

54919–54933. PMLR, 2025. URL https://proceedings.mlr.press/v267/shi25d.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.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/hash/9ee3a664ccfeabc0da16ac6f1f1cfe59-Abstract-Conference.html.

Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1365–1374, 2019. doi: 10.1109/ICCV.2019.00145. URL https://openaccess.thecvf.com/content_ICCV_2019/html/Tung_Similarity-Preserving_Knowledge_Distillation_ICCV_2019_paper.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18638–18646, 2026a. doi: 10.1609/aaai.v40i22.38931. URL https://ojs.aaai.org/index.php/AAAI/article/view/38931.

Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified Vision-Language-Action Model. In *International Conference on Learning Representations*, 2026b. URL https://iclr.cc/virtual/2026/poster/10009646.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998. PMLR, 2022. URL https://proceedings.mlr.press/v162/wortsman22a.html.

Wei Wu, Fan Lu, Yunnan Wang, Shuai Yang, Shi Liu, Fangjing Wang, Qian Zhu, He Sun, Yong Wang, Shuailei Ma, et al. A pragmatic VLA foundation model. *arXiv preprint arXiv:2601.18692*, 2026. doi: 10.48550/arXiv.2601.18692. URL https://arxiv.org/abs/2601.18692.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A. Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1644c9af28ab7916874f6fd6228a9bcf-Abstract-Conference.html.

Senqiao Yang, Chengyao Wang, Yuxin Chen, Zixuan Wang, Longxiang Tang, Haokun Gui, Jinhui Ye, Changsheng Lu, Xiaoyang Wu, Mingkang Zhu, Pengguang Chen, Shu Liu, Zhuotao Tian, Hengshuang Zhao, Bei Yu, and Jiaya Jia. Beyond Data Scaling: Representation-Centric Continued Pre-training for Vision-Language-Action Models. *arXiv preprint arXiv:2608.27550*, 2026a. URL https://arxiv.org/abs/2608.27550.

Yandan Yang, Shuang Zeng, Tong Lin, Xinyuan Chang, Dekang Qi, Junjin Xiao, Haoyun Liu, Ronghan Chen, Yuzhi Chen, Dongjie Huo, Feng Xiong, Xing Wei, Zhiheng Ma, and Mu Xu. ABot-M0: VLA foundation model for robotic manipulation with action manifold learning. *arXiv preprint arXiv:2602.11236*, 2026b. URL https://arxiv.org/abs/2602.11236.

Bin Yu, Shijie Lian, Xiaopeng Lin, Yuliang Wei, Zhaolong Shen, Changti Wu, Yuzhuo Miao, Xinming Wang, Bailing Wang, Cong Huang, and Kai Chen. TwinBrainVLA: Unleashing the potential of generalist VLMs for embodied tasks via asymmetric mixture-of-transformers. *arXiv preprint arXiv:2601.14133*, 2026. doi: 10.48550/arXiv.2601.14133. URL https://arxiv.org/abs/2601.14133.

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.

Ying Zhang, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4320–4328, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_Deep_Mutual_Learning_CVPR_2018_paper.html.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, Tai Wang, Ya-Qin Zhang, Jingjing Liu, and Xianyuan Zhan. X-VLA: Soft-Prompted Transformer as Scalable Cross-Embodiment Vision-Language-Action Model. In *International Conference on Learning Representations*, 2026. URL https://iclr.cc/virtual/2026/poster/10007740.

Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé, III, Andrey Kolobov, Furong Huang, and Jianwei Yang. TraceVLA: Visual Trace Prompting Enhances Spatial-Temporal Awareness for Generalist Robotic Policies. In *International Conference on Learning Representations*, pp. 54277–54296, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/8667f264f88c7938a73a53ab01eb1327-Abstract-Conference.html.

Linqing Zhong, Yi Liu, Yifei Wei, Ziyu Xiong, Si Liu, and Guanghui Ren. ACoT-VLA: Action Chain-of-Thought for Vision-Language-Action Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8152–8162, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zhong_ACoT-VLA_Action_Chain-of-Thought_for_Vision-Language-Action_Models_CVPR_2026_paper.html.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

A Experimental Protocols and Configurations

A.1 Experimental Setup

A.1.1 Benchmarks and Training Setup

Benchmarks and data. The simulation evaluation covers LIBERO (Liu et al., 2023) and RoboTwin 2.0 (Chen et al., 2025). LIBERO includes the Spatial, Object, Goal, and Long task suites under a joint action-learning setup; RoboTwin evaluates bimanual manipulation. Training and evaluation are organized separately for each benchmark, with closed-loop success rate on the corresponding task set as the main metric.

Models and interfaces. The pretrained vision-language backbones use the vision-language components of Qwen3.5-0.8B (Qwen Team, 2026), Cosmos-Reason2-2B (NVIDIA, 2026), and ABot-M0 (Yang et al., 2026b). We retain their first 14, 16, and 22 language layers, respectively. Each branch has 64 action queries and maps queries and contexts into a shared 2048-dimensional space. **Table A.1** lists the corresponding model and interface sizes.

Backbone parameter counts are measured for the actual modules after truncating language layers and adding action-query inputs, including the adjusted word embeddings. Separate learnable action queries, dimension projections, and branch-local adapters are counted separately. The Cosmos branch uses identity dimension mappings; the Qwen and ABot branches project queries and contexts separately. Each branch-local adapter has one cross-attention layer. Action-Centric MoT uses two fusion layers, each with eight attention heads and a feed-forward dimension four times the shared dimension. Action prediction uses a shared DiT flow-matching action head.

Table A.2 summarizes the unified interface stages.

Training configuration. We used AdamW with a backbone learning rate of 10^{-5} and a learning rate of 10^{-4} for queries, projections, adapters, the fusion module, and the action head. LIBERO training comprised 50,000 updates with learning-rate warmup and cosine decay to a minimum of 10^{-6}; RoboTwin training comprised 500,000 updates. Both used eight GPUs with a batch size of eight per GPU. The momentum parameters were $(0.9, 0.95)$; the stability term and weight decay were both 10^{-8}.

Gradient accumulation is set to one step, so eight GPUs with eight samples each yield an effective data batch size of $8 \times 8 = 64$. The flow-matching repetition count R specifies the number of noise and time samples per demonstration example and does not change this data batch size.

Joint supervision. **Table A.3** gives the parameter-update scope of the training losses.

Comparison groups. An independently trained baseline from each source has the same backbone, query interface, and adapter as the corresponding MMA branch, but trains its own action head with the same architecture. An MMA branch is obtained through joint training and executes only that branch and the shared action head at inference. The MMA fusion policy executes all sources and the fusion module.

Ablation definitions. The full-MMA rows in Table 4 use the results of Tables 1 and 2. No query distillation sets $\lambda_{\mathrm{D}} = 0$ while retaining MoT and Pool. Mean query fusion replaces MoT and Pool with the slotwise mean of adapted branch queries and retains distillation. No relational distillation sets $\alpha_{\mathrm{r}} = 0$ while retaining cosine alignment. All settings use $Q = 64$ queries.

A.1.2 Representation and Mechanism Analysis Setup

The representation and transfer analyses used Qwen3.5-0.8B, Cosmos-Reason2-2B, and ABot-M0, with 14, 16, and 22 retained layers; we refer to this as the canonical analysis configuration. Eight held-out LIBERO-90 tasks supplied 32 demonstrations and 96 observations. Diagnostics compared initialization with an intermediate trained checkpoint; the query-spectrum analysis used a separate 10,000-step checkpoint. Flow-error comparisons shared eight noise and time samples within each

Table A.1: Model and interface configurations. Each pretrained source uses its vision-language component. Language layers are reported as retained/original layers; d_k is the native hidden dimension, D the shared dimension, and Q the number of action queries.

Source	Pretrained model	Language layers	d_k	D	Q
Qwen3.5	Qwen3.5-0.8B	14/24	1024	2048	64
Cosmos-Reason2	Cosmos-Reason2-2B	16/28	2048	2048	64
ABot-M0	ABot-M0	22/36	2560	2048	64

Table A.2: Branch inputs and outputs in the unified interface.

Stage	Input	Output
Source feature extraction and splitting	Image, instruction, and learnable queries	Queries $B \times Q \times d_k$; context $B \times T_k \times d_k$
Shared-dimension mapping	Native queries and context	Queries $B \times Q \times D$; context $B \times T_k \times D$
Branch-local adaptation	Mapped queries, context, and mask	$\mathbf{Z}_k \in \mathbb{R}^{B \times Q \times D}$
Shared action prediction	$\mathbf{Z}_k$ and action-generation conditions	$H \times d_a$ action chunk per sample

Context masks exclude padding; all branches use the same action head.

observation, and actions were normalized using training-set statistics. Figure captions specify the relevant model state.

A.1.3 Inference Cost Measurement

The deployment measurements in Table 5 and Figure 6 cover Qwen3.5-0.8B, Cosmos-Reason2-2B, and ABot-M0. Hardware, numerical precision, camera inputs and resolution, batch size, action horizon, sampling steps, warmup, and timing repetitions were fixed across policies. Median latency runs from in-memory PIL views to CPU normalized actions; video decoding, input resizing, RPC, and environment execution are excluded. Peak allocated GPU memory and resident parameter counts cover deployed components only: one branch and the shared action head, or all sources, fusion, and the shared head. The figure decomposes mean exclusive time by inference stage.

A.1.4 Real-World Task Protocol

We evaluated on a tabletop robot arm with four block-manipulation tasks (**Figure A.1**): pick and place three blocks into a box, replace a red block with a blue block, arrange red, green, and blue blocks from left to right, and stack three blocks into a tower. These tasks cover pick-and-place, color discrimination, spatial ordering, and stacking. We collected 100 demonstrations per task. At test time, each task was repeated 20 times to measure success rate.

Table A.3: Losses and parameter updates.

Loss	Branch parameters θ_k	Fusion parameters ψ	Shared action head ϕ
Action loss of branch k	Updates branch k	No update	Updates
Fusion action loss	Updates all participating branches	Updates	Updates
Query distillation of branch k	Updates student branch k	No update	No update

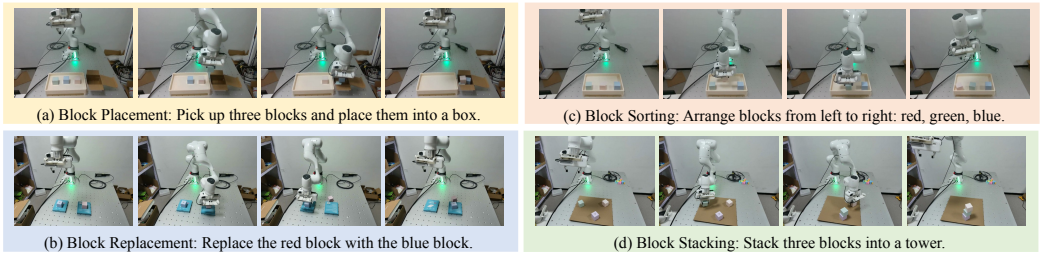


Figure A.1: Examples of real-world tasks. (a) Block placement. (b) Block replacement. (c) Block ordering. (d) Block stacking. Each image sequence shows the manipulation from left to right.

B Additional Representation and Transfer Analyses

Representation diagnostics use fixed held-out observations and instructions to extract adapted branch queries $\mathbf{Z}_k$ and pooled fused queries $\mathbf{Z}_{\mathrm{F}}$, normalized as in Eq. 7. Corresponding-query similarity is computed between student and teacher in the same jointly trained model; similarity distributions are compared across checkpoints. Relational matrices use the teacher and student distributions in Eq. 9, both referenced to fused teacher queries and plotted with a consistent color scale.

Our checkpoint comparisons use paired observations from initialization and an intermediate trained checkpoint to measure changes in action-probe error (**Figure B.2**), query alignment, and flow-velocity error (**Figure B.8**).

B.1 Representation Learning and Training Changes

We compare saved initialization with an intermediate trained checkpoint through module parameter displacement and linear readout of action information.

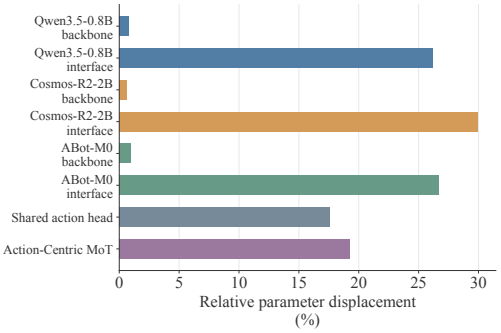


Figure B.1: Net parameter displacement by module from initialization to an intermediate trained checkpoint in the canonical configuration. The ratio is the L2 norm of the parameter difference divided by the L2 norm at initialization, using the BF16-rounded initialization from training startup. The interface includes action queries, dimension mapping, and branch adapters.

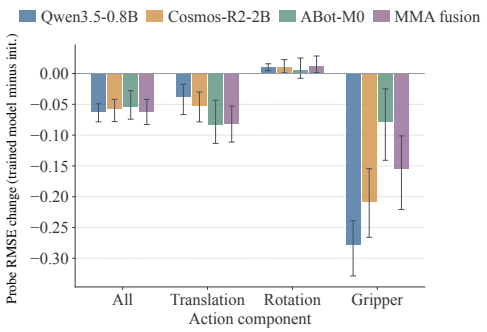


Figure B.2: Change in linear-probe error for action components between initialization and an intermediate trained checkpoint. Inputs are flattened 64×2048 queries, with nested leave-one-scene-cluster-out validation over seven clusters. Bar heights are paired RMSE differences (checkpoint minus initialization); negative values indicate lower error. The 95% intervals come from paired resampling of scene clusters using fixed outer-fold predictions.

In **Figure B.1**, relative module displacement ranges from approximately 0.67% for the Cosmos backbone to 29.95% for the Cosmos interface. The interface and backbone change by different amounts during joint training. This statistic measures net change between saved initialization weights and an intermediate checkpoint.

Figure B.2 shows component-level differences in how action information is linearly read out at initialization and the intermediate checkpoint. Translation and gripper probe errors decrease, while the rotation point estimate increases. The Qwen gripper change is about -0.27886, and the fusion rotation change is about 0.01283. The rotation-change intervals for Qwen, Cosmos, and fusion lie above zero; the ABot-M0 interval crosses zero.

B.2 Query Structure

Figure B.3 examines covariance across query slots at the 10,000-step checkpoint of the Qwen–Cosmos–ABot configuration.

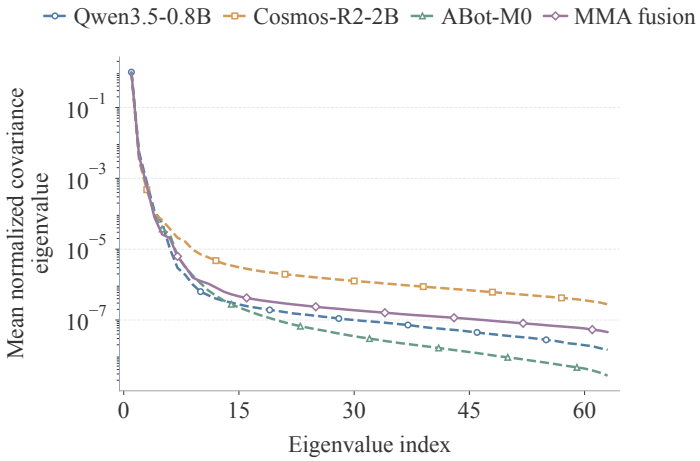


Figure B.3: Query covariance spectrum at 10,000 steps. Queries are centered across 64 slots within each observation. Eigenvalues of normalized covariance are averaged over 96 observations; the vertical axis shows energy share on a logarithmic scale.

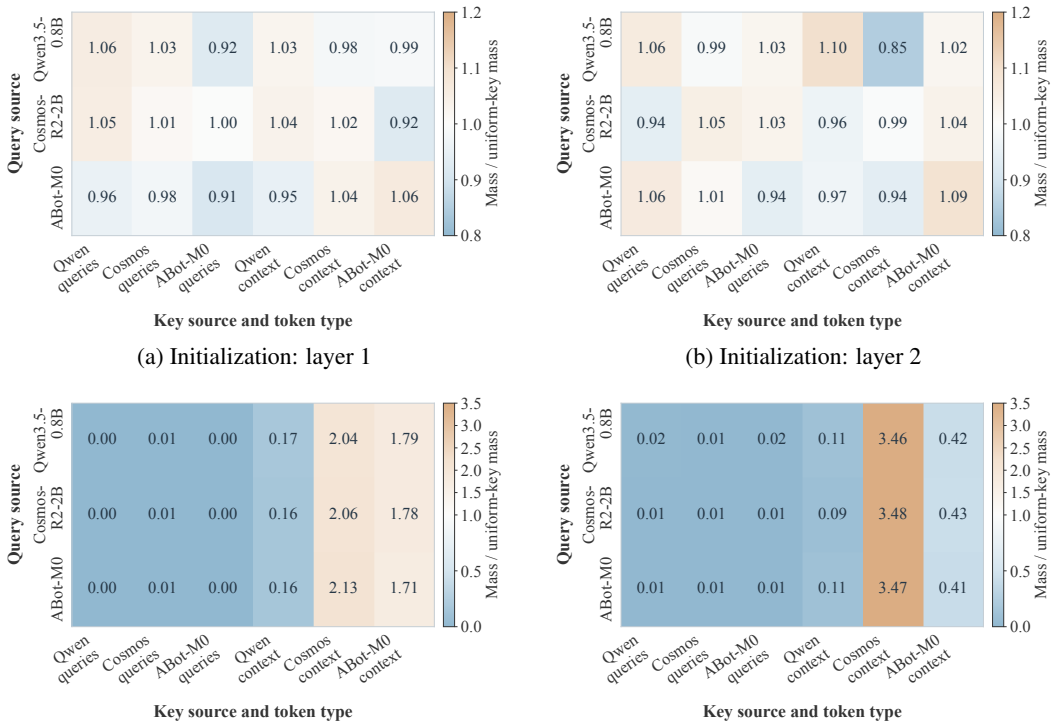
Slot covariance energy is concentrated in the first spectral component, whose mean share is about 0.996 for Cosmos. Other components decay rapidly, indicating that most within-observation query variation lies in a few directions.

B.3 Training Changes and Interventions in Fusion

We compare source reading at initialization and an intermediate trained checkpoint, then intervene on queries and contexts to examine output responses.

Figure B.4 compares initialization and an intermediate trained checkpoint on the same held-out panel. Initial enrichment ranges from approximately 0.849 to 1.101, close to uniform per-key reading. At the checkpoint, the range expands to approximately 0.004–3.477, with more concentrated allocation across sources and token types.

Figure B.5 shows error direction, output response, and representation change under source interventions. Masking ABot context changes flow MSE by approximately -0.00621, with a 95% interval of $[-0.01255, -0.00078]$; error decreases on this held-out panel. Query and context replacements also produce different magnitudes of velocity response and fused-query change, showing that both interface inputs affect fused output.



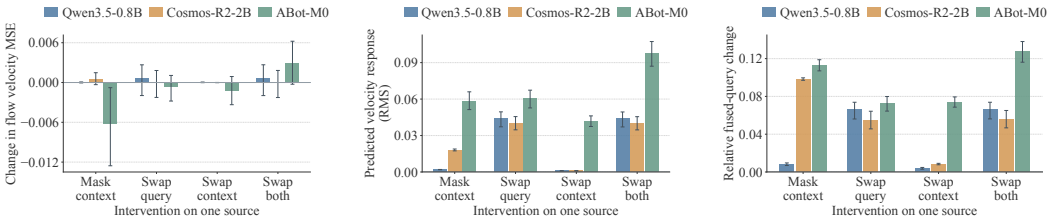
(a) Initialization: layer 1

(b) Initialization: layer 2

(c) Trained model: layer 1

(d) Trained model: layer 2

Figure B.4: Changes in source reading in the canonical configuration. (a,b) Layers 1 and 2 at initialization. (c,d) Layers 1 and 2 at an intermediate trained checkpoint. For 96 fixed held-out observations, attention heads and queries are averaged first, then tasks are weighted equally. Colors are centered at enrichment 1; the two model states use the separate color scales shown in the plots.



(a) Change in flow MSE

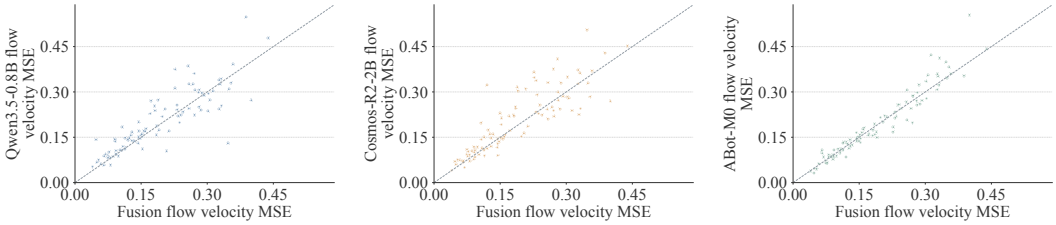
(b) RMS of predicted-velocity response

(c) Relative change in fused queries

Figure B.5: Source interventions at an intermediate trained checkpoint of the canonical model. (a) Change in flow MSE. (b) RMS of predicted-velocity response. (c) Relative change in fused queries. We mask context, replace queries, replace context, or replace both. Masking retains the source queries; replacement uses another demonstration of the same task at the corresponding relative position. The 96 observations share flow conditions, tasks are weighted equally, and error bars are 95% intervals from 10,000 paired resamples of seven scene clusters.

B.4 Teacher–Branch Prediction Relationships and Knowledge Transfer

This section examines the query-distillation objective in Section 3.4. We compare teacher and branch prediction errors, corresponding-query directions, and anchor-based relational distributions, then analyze paired changes from initialization to an intermediate trained checkpoint.



(a) Qwen3.5-0.8B (b) Cosmos-Reason2-2B (c) ABot-M0

Figure B.6: Paired teacher and branch errors at an intermediate trained checkpoint of the canonical model. (a) Qwen. (b) Cosmos. (c) ABot-M0. Eight held-out tasks, 32 demonstrations, and 96 observations are used; each point averages eight shared flow conditions. Horizontal and vertical axes show fusion and branch MSE, respectively. All panels use the same axis range; the dashed line is $y = x$.

Figure B.6 compares teacher and branch on the same observations and flow conditions. Points above the diagonal indicate lower error for the fusion teacher; points below indicate lower branch error. Error differences vary across sources, and the teacher advantage depends on the observation and source.

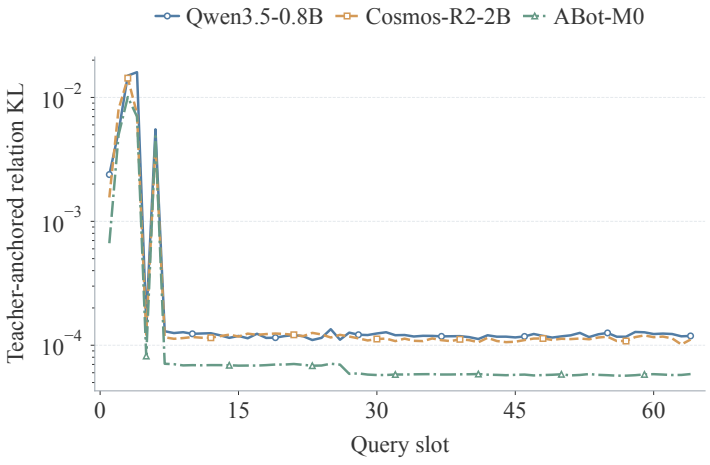
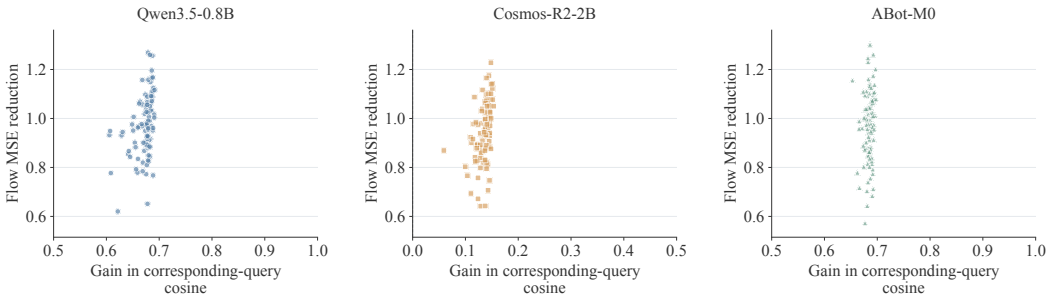


Figure B.7: Teacher-anchored relational KL by query slot at an intermediate trained checkpoint. At each slot, teacher-to-student KL between anchor-based relational distributions is computed as in Eq. 10, then averaged over 96 observations. The vertical axis uses a logarithmic scale.

Figure B.7 shows how relational KL varies across query slots. Branch curves have different local fluctuations across slots.



(a) Qwen3.5-0.8B (b) Cosmos-Reason2-2B (c) ABot-M0

Figure B.8: Increase in alignment and decrease in flow error from initialization to an intermediate trained checkpoint. (a) Qwen. (b) Cosmos. (c) ABot-M0. The horizontal axis shows the increase in corresponding teacher-query cosine similarity, and the vertical axis shows flow MSE at initialization minus that at the checkpoint, using the same 96 paired observations. Horizontal-axis spans are equal across panels, with branch-specific starting points.

Figure B.8 shows increased directional alignment and decreased flow error for all three branches from initialization to the intermediate checkpoint. Alignment increases are larger for Qwen and ABot than for Cosmos. The paths show different magnitudes of representation change under joint action supervision and query distillation.