Joint-Aligned Latent Action: Towards Scalable VLA Pretraining in the Wild

Supplementary Material

A. Implementation: Pre-Training on Human Videos

A.1. Motion Tokenization

Our JALA architecture follows the Being-H0 design and is built on InternVL3-2B as the vision-language backbone with 28 attention layers. To incorporate the motion modality, we tokenize *15-frame motion chunks*. Each chunk is decomposed into *wrist* and *finger* motions, which are separately quantized into 64 tokens each, yielding 128 tokens per chunk. The codebook size of each part is 4096, learned via *GRVQ* [66], a vector quantization algorithm, to capture general motion patterns. The motion tokens are then wrapped with two **special tokens** `<mot>` and `</mot>`, forming the following unified format:

`<mot>` {wrist_motion_tokens} {finger_motion_tokens} `</mot>`

These special tokens serve as explicit delimiters, helping the VLM distinguish motion chunks from other modalities such as vision or instruction tokens. For sequences containing **both hands**, we interleave the left-hand and right-hand chunks along the temporal axis to preserve synchrony while maintaining modality distinction.

A.2. Latent Action Perceiver (LAP) & Latent State Perceiver (LSP)

Structure. The Latent Action Perceiver (LAP) and Latent State Perceiver (LSP) share the same structure and weights. We adopt a 2-layer Perceiver module, where each layer consists of:

1. A **cross-attention block**, using visual features as key-value pairs and learnable latent queries for information extraction.
2. A **self-attention block** applied on the extracted latent tokens for within-latent aggregation.

The perceiver modules process visual features from texture-perceptive visual encoders (such as DINO, V-JEPA, and so on) into action/state latent, which are further passed through a 2-layer MLP to map them into the VLM embedding space. Since videos may contain two hands, latent actions originating from the same Perceiver might be aligned to different motion chunks. To maximize sharing while allowing differentiation, we double the channel dimension of the Perceiver's MLP and split it into two hand-specific heads. Each sample dynamically selects the left- or right-hand head depending on the active motion stream. Specially, for consistency between the Latent Action Perceiver (LAP) and the Latent State Perceiver (LSP), both modules are designed to process pairs of frames. While LAP takes the boundary frames $(v_t, v_{t+\delta})$ of each motion chunk, in LSP we duplicate the initial frame (v_0, v_0). This ensures that differences between LAP and LSP stem solely from their input semantics (dynamics vs. context), rather than from architectural mismatch.

EMA update for Perceiver. A key challenge in aligning predictive embeddings with latent actions is that the Latent Action Perceiver (LAP) and Latent State Perceiver (LSP) process heterogeneous signals (visual boundary frames vs. predictive context). Direct joint optimization often leads to instability or collapse. To stabilize training, we adopt a decoupled exponential moving average (EMA) update between the two modules. Concretely, let $\theta_b^{\text{LAP}}, \theta_q^{\text{LAP}}$ denote the backbone and query parameters of LAP, and $\theta_b^{\text{LSP}}, \theta_q^{\text{LSP}}$ those of LSP. We update them as:

$$\theta_b^{\text{LAP}} \leftarrow \alpha\, \theta_b^{\text{LAP}} + (1-\alpha)\, \theta_b^{\text{LSP}}, \tag{6}$$

$$\theta_q^{\text{LSP}} \leftarrow \alpha\, \theta_q^{\text{LSP}} + (1-\alpha)\, \theta_q^{\text{LAP}}, \tag{7}$$

where $\alpha \in [0, 1)$ is the EMA coefficient. This asymmetric design ensures that the LAP backbone stays consistent with the predictive context shaped by MCP and alignment losses, while the LSP queries gradually inherit the action-grounding capability from LAP. In practice, we set $\alpha = 0.999$ to balance stability and adaptability.

A.3. Masked Chunk Prediction (MCP)

Hybrid Masking Scheme. During pretraining, we apply masked decoding to enforce predictive consistency. A naive masking strategy introduces a mismatch between training and inference, since all tokens in a chunk are replaced by `[MASK]` during training, but in inference, motion chunks are generated sequentially. To mitigate this gap, we employ a **hybrid masking scheme**:

- For each sequence with N chunks, we randomly select one chunk as the main prediction target.
- Chunks before the target are kept intact (no masking).
- Inside the target chunk, each token is masked with a random ratio uniformly sampled from $\{0.05, 0.15, \ldots, 1.0\}$.
- Tokens in chunks after the target are masked with a fixed 5% probability to provide additional supervision without distorting context.

This ensures that the main prediction chunk processes aligned context during both training and inference.

Full Masking on Unlabeled videos. For in-the-wild *video-only* samples that lack motion tokens, the entire motion chunk A_i is replaced by `[MASK]` placeholders. In this case, the MCP term is inactive, and training proceeds solely via alignment to latent actions from LAP (i.e., only $\mathcal{L}_{\text{Align}}$ is applied). This keeps the interface unified while still learning predictive embeddings that are aligned to dynamics without requiring explicit motion labels.

Inference with MCP. For motion generation, we decode the current chunk **multiple times**, each time decoding $\sim$5% of the tokens in the chunk. Finally, the outputs are ensembled to reduce approximation error. This retains the efficiency advantage over causal decoding, while downstream transfer still uses a **single forward pass** to extract predictive embeddings.

A.4. Training Details

We optimize JALA with AdamW using a base learning rate of 3×10^{-5}, weight decay of 0.05, and $\beta = (0.9, 0.95)$. The learning rate is warmed up for the first 5% steps and then decayed with a cosine schedule, while gradient clipping (max norm 1.0) is applied throughout. Training uses an effective batch size of 128 sequences, obtained from a per-GPU batch size of 16 with gradient accumulation across 8 GPUs, where each sequence contains a 15-frame motion chunk plus paired instructions and boundary frames. The hybrid loss combines masked chunk prediction and latent-action alignment with $\lambda = 0.5$, and for in-the-wild videos, only the alignment loss is applied. The perceiver modules are updated with an EMA coefficient of $\alpha = 0.999$ to stabilize training. Pretraining is performed on the full 7.5M UniHand-Mix dataset for a single epoch, which requires 68 hours on 8 NVIDIA A100 (80GB) GPUs.

A.5. Hand Motion Generation Evaluation

Dataset Setups. We evaluate JALA's hand-motion generation on two distinct splits of the UniHand-Mix dataset, following the same protocol used for motion modeling: (1) **Lab split**, a held-out subset of lab-annotated sequences measuring fidelity under precise supervision; and (2) **Wild split**, curated from Ego4D videos with HaWoR annotations, emphasizing generalization to unseen and in-the-wild manipulation behaviors. The two splits reflect different operating regimes: controlled indoor environments versus diverse, unconstrained real-world human activities.

Evaluation Metrics. We employ four metrics to comprehensively evaluate JALA's ability to generate spatially accurate, temporally coherent, and instruction-faithful hand motions:

- **MPJPE (Mean Per Joint Position Error).** Measures spatial accuracy by computing the mean Euclidean distance between each predicted 3D hand joint and its ground-truth counterpart over the entire sequence.
- **PA-MPJPE (Procrustes Aligned MPJPE).** Evaluates relative pose fidelity by rigidly aligning the predicted joints to the ground truth (rotation, translation, and scaling) before computing MPJPE.
- **MWTE (Mean Wrist Translation Error).** Assesses global trajectory fidelity by computing the mean Euclidean distance between predicted and ground-truth wrist positions across the sequence, capturing long-horizon wrist motion accuracy.
- **MDE (Mean Direction Error).** Measures consistency of motion trends by comparing the final wrist displacement direction relative to the initial wrist position between prediction and ground truth.

These metrics jointly capture both local pose-level fidelity and global trajectory realism of generated hand motions across controlled (Lab) and in-the-wild (Wild) evaluation conditions.

B. Implementation: Post-Training on Robot Manipulation Tasks

B.1. Down-stream Adaptation Mechanism

Flow-matching head for robot adaptation. For post-training adaptation, we directly follow the flow-matching head design in GR00T N1.5 [5]. Given predictive embeddings $\{h_{i,k}\}$ from the pretrained VLA backbone, we use them as conditional input to a Diffusion Transformer (DiT) policy head composed of alternating self-attention and cross-attention layers. We employ a DiT with 16 layers of 32-head attention blocks, and the hidden state dimension is 2048. Self-attention operates over the robot's proprioceptive states and noisy actions, while cross-attention integrates predictive embeddings $\{h_{i,k}\}$ to inject human-derived dynamics knowledge. The training objective is a flow-matching loss:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{\tau,\epsilon,A_t}\left[\|V_\theta(\{h_{i,k}\}, A_t^\tau, q_t) - (\epsilon - A_t)\|_2^2\right], \tag{8}$$

where $A_t^\tau = \tau A_t + (1-\tau)\epsilon$ is a noised action chunk, q_t is the proprioceptive state, and $\epsilon \sim \mathcal{N}(0, I)$. During inference, we initialize with Gaussian noise and iteratively denoise using forward Euler steps, typically with a fixed schedule of N steps (we use $N = 4$ by default). This process generates precise action sequences consistent with the pretrained latent action space, enabling efficient transfer of dynamics-rich human priors to robotic control tasks.

Training Details. For downstream adaptation with the flow-matching head, we freeze the visual modules and use a batch size of 128, learning rate of 1×10^{-4} with cosine decay and 5% warm-up. We run 30k steps (≈8 hours) on LIBERO tasks, while we run 60k steps (≈16 hours) on RoboCasa tasks, using the same hardware configuration as that in the pretraining stage.

B.2. Real-World Robot Setting

Robot System. We use a real-robot setup consisting of a 7-DoF Franka Research 3 arm, a 6-DoF Inspire dexterous hand, and Intel RealSense L515 depth cameras for third-person observations. The policy receives third-person RGB observations and predicts action chunks that are executed by the arm–hand system.

Manipulation Tasks. We evaluate JALA on three multi-step manipulation tasks designed to probe spatial reasoning, long-horizon control, contact handling, and cross-object generalization:

- **`Put-Three-Obj.`** The robot must open a drawer, place three fruits inside, and then close the drawer. This task stresses multi-object coordination, precise pose estimation, and long-horizon manipulation. It is decomposed into *five* subtasks: (1) open the drawer, (2) pick and place fruit #1, (3) pick and place fruit #2, (4) pick and place fruit #3, (5) close the drawer.
- **`Wipe-Board.`** The robot wipes pen marks from a whiteboard using a cloth. This task requires sustained contact, continuous wiping motion, and region-aware perception under occlusion. Following the data-collection setup, we use *two* camera views to mitigate occlusion caused by the arm. The task consists of *three* subtasks: (1) grasp and lift the cloth, (2) perform wiping motions over the marked region, (3) successfully remove the majority of visible ink.
- **`Water-Plant.`** The robot grasps a spray bottle, moves it to the plant, and activates the trigger to water the plant. The task evaluates spatial reasoning, accurate placement, and fine-grained trigger actuation using the multi-finger Inspire hand. It contains *three* subtasks: (1) grasp the spray bottle, (2) reposition it in front of the plant, (3) activate the trigger to produce a water spray.

Figure 5 shows representative sequences from our teleoperated demonstrations for all three tasks. For each task, we collect 50 teleoperated demonstrations. Each demonstration is aligned with third-person RGB observations and textual instructions for downstream post-training.

For each task and its unseen variant, we execute 10 rollouts and evaluate performance with **Completion Ratio**, which is defined as the proportion of subtasks successfully completed within a rollout. To assess robustness under visual shifts: (1) the tablecloth pattern is changed in `Put-Three-Obj`, and (2) the marker color is altered in `Wipe-Board`.

C. Dataset Details

C.1. Data Curation Steps

Lab-Collected Subset. Following the UniHand [39] pipeline, we curate a high-quality lab-collected subset with precise 3D hand annotations and dense task descriptions.

Figure 5. **Teleoperated demonstration examples** for the three real-world manipulation tasks. **Top**: **`Put-Three-Obj`** (open drawer → pick&place three fruits → close drawer). **Middle**: **`Wipe-Board`** (grasp cloth → wipe over marked region → remove visible ink). **Bottom**: **`Water-Plant`** (grasp spray bottle → reposition toward plant → activate trigger).

1. **Hand pose standardization:** All annotations are unified into the MANO parameter format [51]. For datasets with mocap or SLAM labels, we directly convert to MANO. For datasets with only 3D joints, we fit MANO via gradient optimization. For raw RGB-only data, we apply HoWaR [72] for per-frame estimation, followed by temporal smoothing and left–right correction.
2. **Task labeling:** Videos are segmented into 10s chunks and annotated hierarchically. At the chunk level, we produce imperative instructions and concise summaries. At the second level, we annotate contact states, object properties, and hand–object interactions, including both two-handed and single-hand actions.
3. **Instructional data generation:** Based on these annotations, we construct multimodal tasks—motion generation, motion translation, and motion prediction—using 20 base templates per task type, diversified via Gemini. This establishes explicit grounding between vision, language, and motion.

In-the-Wild Subset. To complement lab data with naturalistic human behaviors, we curate an additional subset from Ego4D [20]. Unlike controlled setups, egocentric recordings bring challenges such as frequent hand occlusion and irrelevant non-manipulative segments. To extract meaningful manipulation episodes, we employ a two-stage filtering pipeline. First, **visual filtering:** hand regions are detected with an off-the-shelf detector [48], and candidate clips are annotated with HaWoR [72] to obtain approximate pose supervision. Only high-confidence clips are retained, and the threshold is 0.65 for both. Second, **instruction validation:** `Gemini-2.5-Flash` is prompted to verify the presence of hand-centric activities,

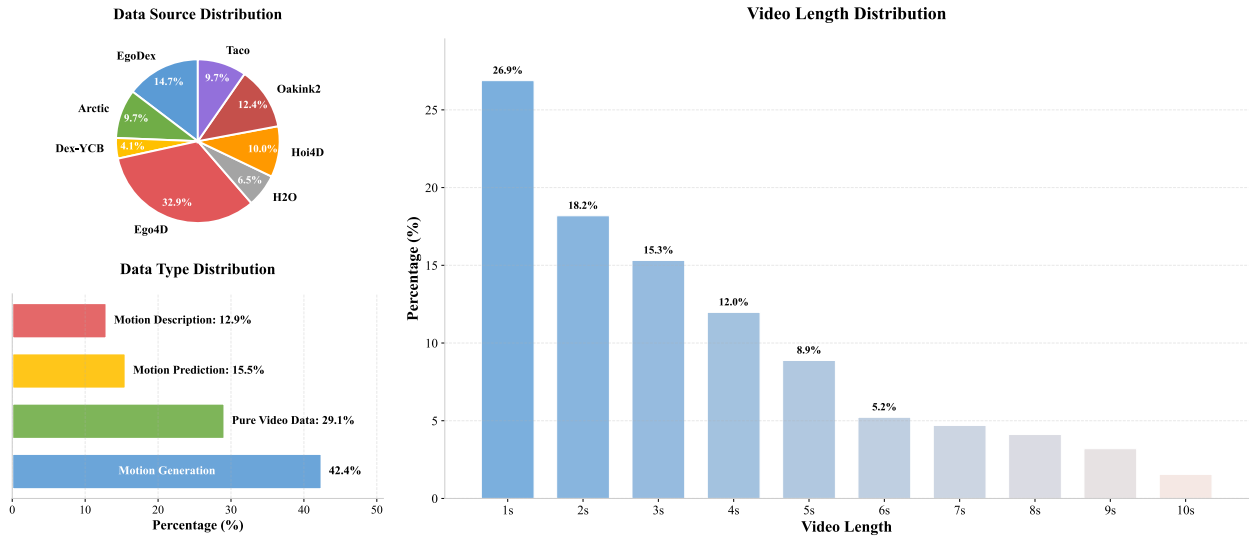

Figure 6. **Dataset statistics of UniHand-Mix.** (Left top) Source distribution across lab-collected and in-the-wild datasets. (Left bottom) Breakdown by task type. (Right) Distribution of clip lengths (1–10 seconds). Together, these statistics highlight the diversity and scale of UniHand-Mix, enabling scalable pretraining.

removing idle or distractor clips. Valid clips are then paired with automatically generated natural language instructions. This process yields roughly 2.5M instruction–video pairs, which, combined with 5M lab-annotated samples, form the UniHand-Mix dataset. The two subsets provide complementary strengths: lab data anchors physical accuracy, while in-the-wild data brings diversity and contextual richness, together enabling scalable pretraining.

C.2. Dataset Statics

To provide a clearer picture of the curated UniHand-Mix dataset, we summarize its composition and distributional properties in Fig. 6. The dataset combines diverse sources, task types, and temporal scales:

- **Data Sources.** As shown in the pie chart (top left), UniHand-Mix integrates both lab-collected datasets (e.g., DexYCB, OakInk2, H2O, Arctic, EgoDex, TACO) and in-the-wild videos (Ego4D), yielding a balanced mix of precise motion annotations and naturalistic behaviors.
- **Data Types.** The bar chart (bottom left) shows the breakdown by task type: motion generation (42.4%), motion prediction (15.5%), motion description (12.9%), and pure video data without explicit motion tokens (29.1%).
- **Video Lengths.** The histogram (right) illustrates the distribution of clip lengths. While short clips (1–3 seconds) dominate, longer sequences (up to 10 seconds) are also included, ensuring coverage of both fine-grained and long-horizon manipulations.

D. Additional Experimental Results

D.1. Discussion: Reconstruction vs. Joint Alignment

Summary of empirical trends. In the main text we observe that JALA consistently outperforms LAPA/LAPA† on two-view LIBERO (Table 3) and shows stronger gains on RoboCasa and GR1 (Table 5), including few-shot settings and embodiment shift.

Reconstruction signal. Reconstruction-based objectives (e.g., LAPA) provide dense pixel supervision, which can place more weight on appearance, background, and camera artifacts in in-the-wild videos. This is not always aligned with action-relevant dynamics, especially when hands are small or partially occluded.

Alignment signal. Joint alignment constrains predictive embeddings using boundary-frame dynamics, so the supervision is more closely tied to motion rather than appearance. This offers a more behavior-centric training signal that transfers better when data are scarce or domain shifts are large (as in RoboCasa/GR1).

Controlled comparison. LAPA† is trained on our data with the same backbone as JALA, isolating the effect of the training objective. The remaining performance gap therefore reflects differences between reconstruction and alignment, rather than backbone or data scale.

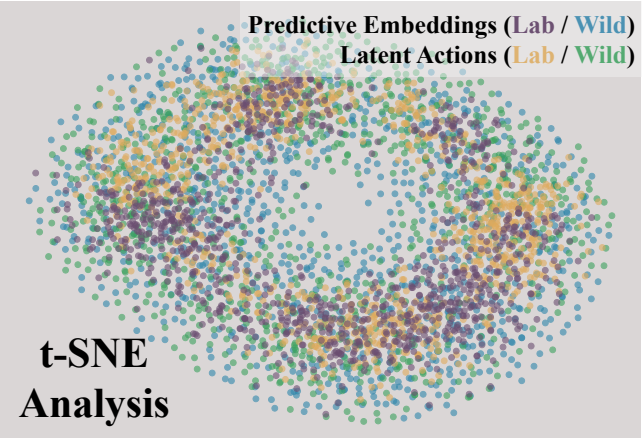

Figure 7. **t-SNE of predictive embeddings h and latent actions z across Lab and Wild.** The two spaces cluster in closely aligned regions, and Wild samples largely expand the Lab manifold, indicating integrated coverage rather than a disjoint domain.

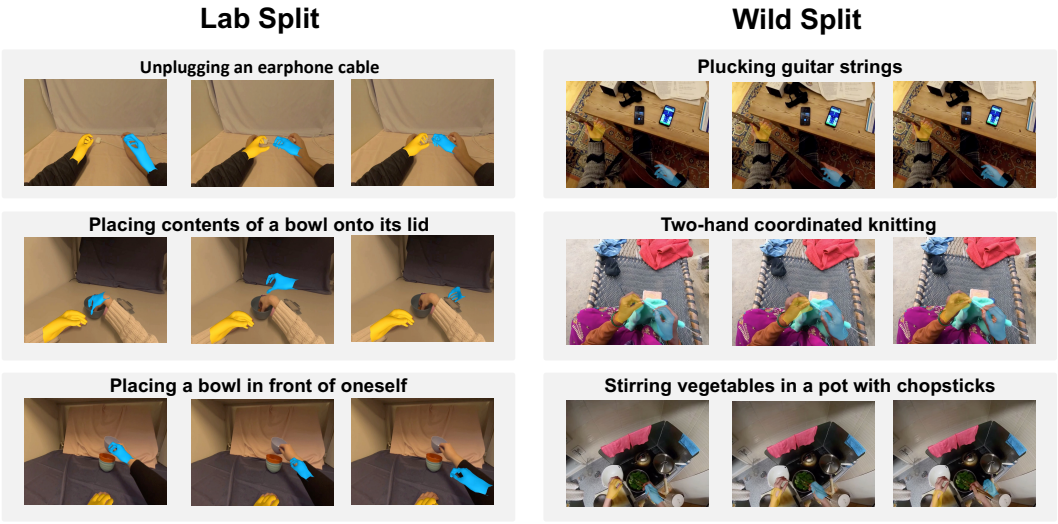

Figure 8. **Qualitative hand-motion generation on lab (left column) and wild (right column) scenes.** Colored overlays denote generated hand poses.

Compute tradeoff. With identical backbone/data, LAPA† still requires two-stage pretraining and longer wall-clock time, while JALA achieves higher accuracy with a single-stage alignment objective. This supports the view that focusing supervision on behavior-relevant dynamics can be more effective than increasing reconstruction compute.

D.2. Latent Alignment Visualization

D.3. Qualitative Hand Motion Generation Examples

Figure 8 presents qualitative hand–motion generations covering both in-the-wild and lab–collected scenarios. On the **wild** side (left column), our model successfully handles diverse, unconstrained interactions such as `plucking guitar strings`, `two-hand coordinated knitting`, and `stirring vegetables in a pot with chopsticks`, demonstrating robust generalization to complex scenes and bimanual coordination. On the **lab** side (right column), the model produces precise and temporally consistent motions for `unplugging an earphone cable`, `placing contents of a bowl onto its lid`, and `placing a bowl in front of oneself`, reflecting accurate fine-grained control in structured settings. These results illustrate that JALA's predictive embeddings support both generalization in the wild and accuracy in controlled environments.

D.4. Analysis of Qualitative Real-Robot Rollouts

To better understand the behavioral competence learned by JALA, we qualitatively analyze high-quality successful rollouts from both the *seen* and *unseen* evaluation settings. Representative sequences are visualized in Figures 9 and 10. Across tasks, these rollouts reveal that the learned policy exhibits not only robust perception and fine-grained manipulation, but also dynamic error recovery and adaptive control strategies typically associated with expert teleoperation.

Put-Three-Obj. In both seen and unseen environments, the policy demonstrates accurate spatial reasoning, precise pre-grasp alignment, and consistent grasp stability across objects of different shapes. A noteworthy behavior emerges in the unseen case: while attempting to grasp the banana, the initial approach is slightly misaligned due to its curved geometry and shifted texture pattern (*see the top sequence in Figure 10*). Instead of committing to an incorrect lift, the policy subtly retracts, re-positions its wrist, and performs a corrected grasp on the second attempt. This self-corrective motion is not explicitly supervised but arises from the latent action modeling, indicating that the policy has internalized a feedback-driven strategy for recovering from minor pose estimation errors. After grasping, the placement motions remain stable and collision-free, even under an altered tablecloth patterns in the unseen domain, demonstrating robustness to visual appearance shifts.

Wipe-Board. Successful rollouts illustrate the policy's ability to maintain firm planar contact with the board and execute smooth, directionally coherent wiping trajectories. A key observation is that the ink is rarely removed in a single stroke. Instead, the policy repeatedly reorients the cloth and revisits regions where residual ink remains (*see the lower sequence in Figures 9 and 10*). This indicates that the model continuously interprets the evolving visual state and adjusts its motion to eliminate remaining marks—a behavior analogous to iterative refinement in human cleaning. Such dynamic adaptation highlights the effectiveness of JALA's perceptual-motor coupling and its ability to handle non-deterministic, contact-rich tasks in a closed-loop manner. Importantly, this behavior persists even when marker colors change in the *unseen* setting, suggesting strong generalization to new visual distributions.

Water-Plant. The policy consistently exhibits a reliable grasp of the spray bottle and a stable reorientation of the container toward the plant. Trigger activation—requiring precise multi-finger coordination—is executed cleanly and without premature release. The entire sequence remains stable, reflecting strong control robustness and precise end-effector modulation.

Overall, these successful rollouts reveal that JALA not only reproduces teleoperated demonstrations but also learns competent manipulation strategies that combine fine-grained control, online visual feedback integration, error recovery, and robustness to unseen environmental variations.

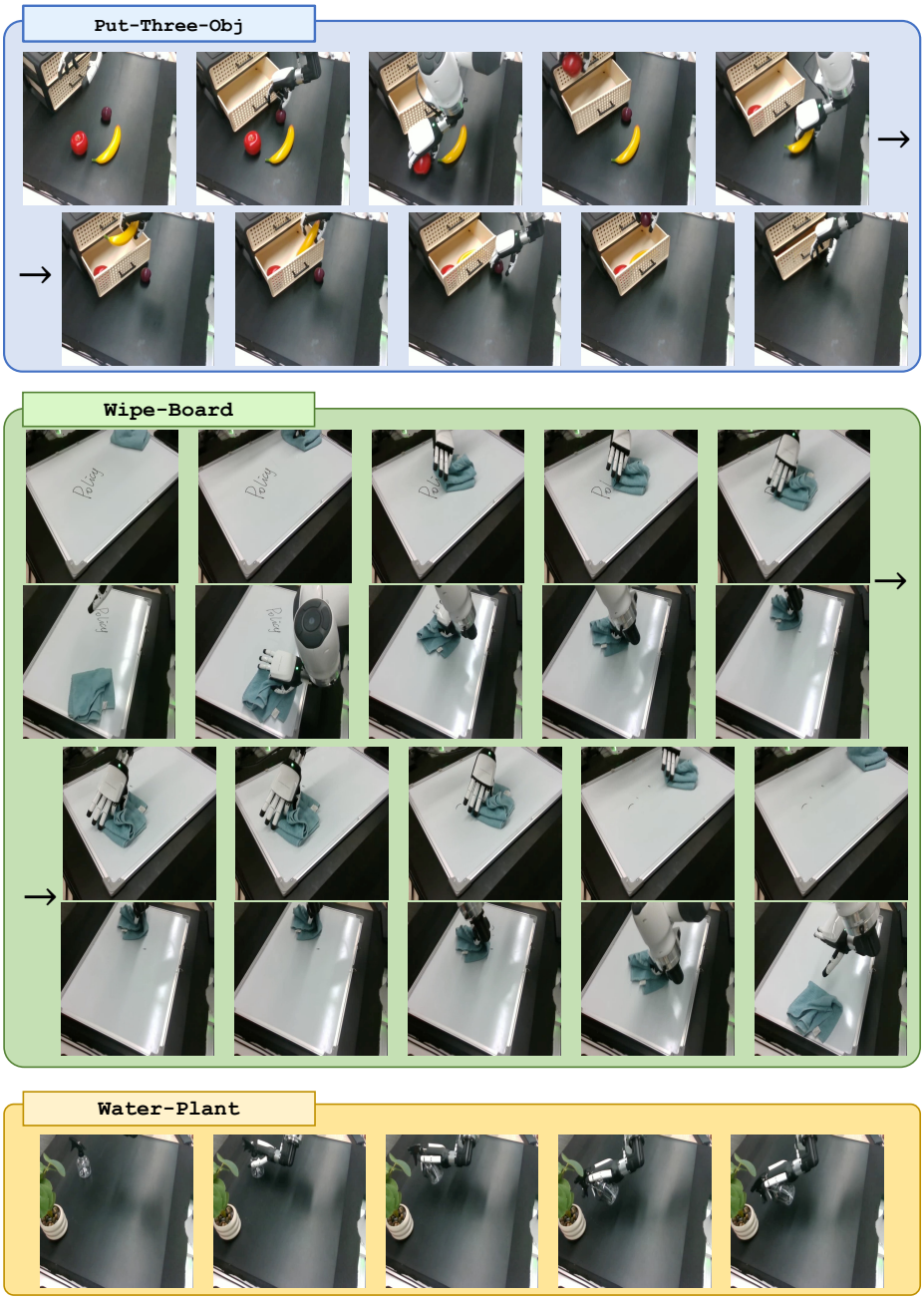

Figure 9. **Successful real-robot rollouts (seen setting).** Top: `Put-Three-Obj`—the policy performs precise grasping and stable placement over the full five-step sequence. Middle: `Wipe-Board`—the policy iteratively wipes regions with residual ink, dynamically adjusting trajectories based on visual feedback. Bottom: `Water-Plant`—the policy reliably grasps, repositions, and actuates the spray bottle with multi-finger control.

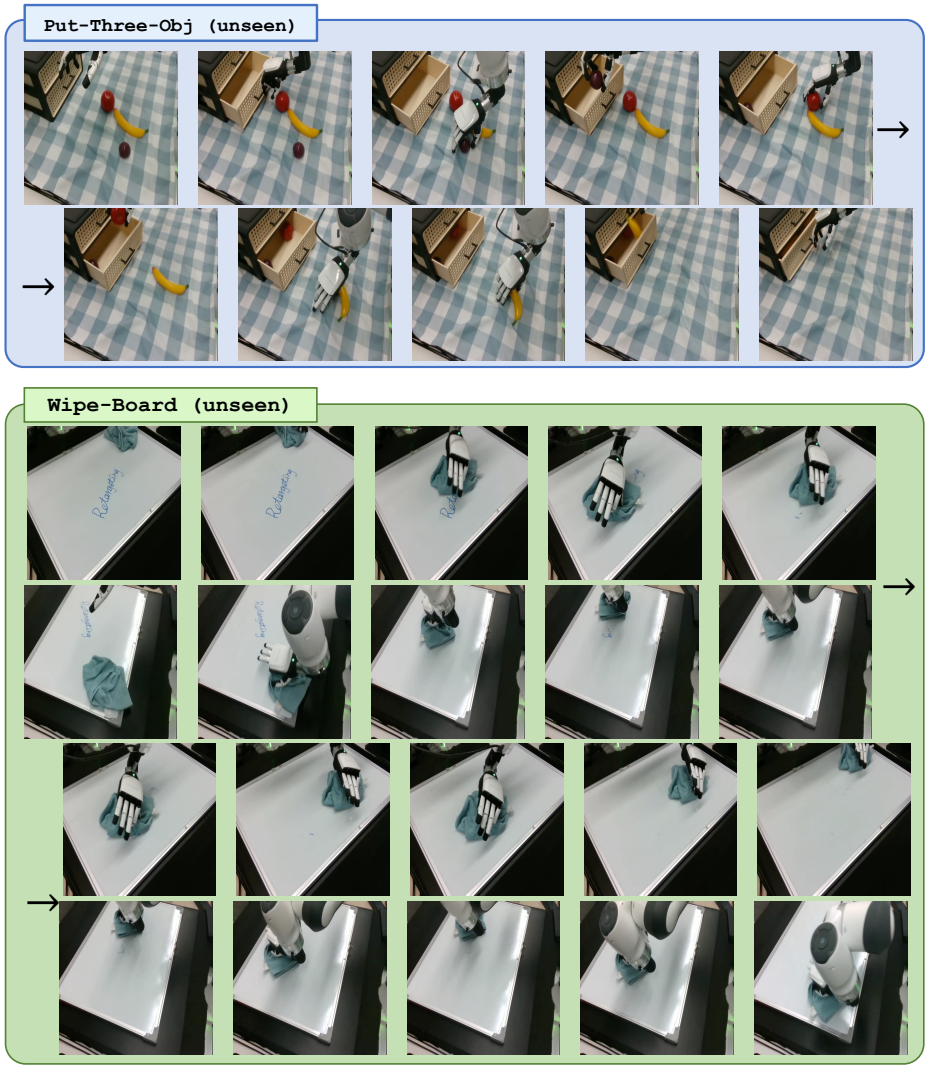

Figure 10. **Successful real-robot rollouts (unseen setting).** `Put-Three-Obj`: The altered tablecloth texture induces initial grasp misalignment for the banana; the policy autonomously retracts and corrects its grasp. `Wipe-Board`: With a changed marker color, the policy continues to adaptively revisit areas with residual ink until the surface is clean. These behaviors demonstrate strong robustness to appearance shifts and dynamic feedback control.

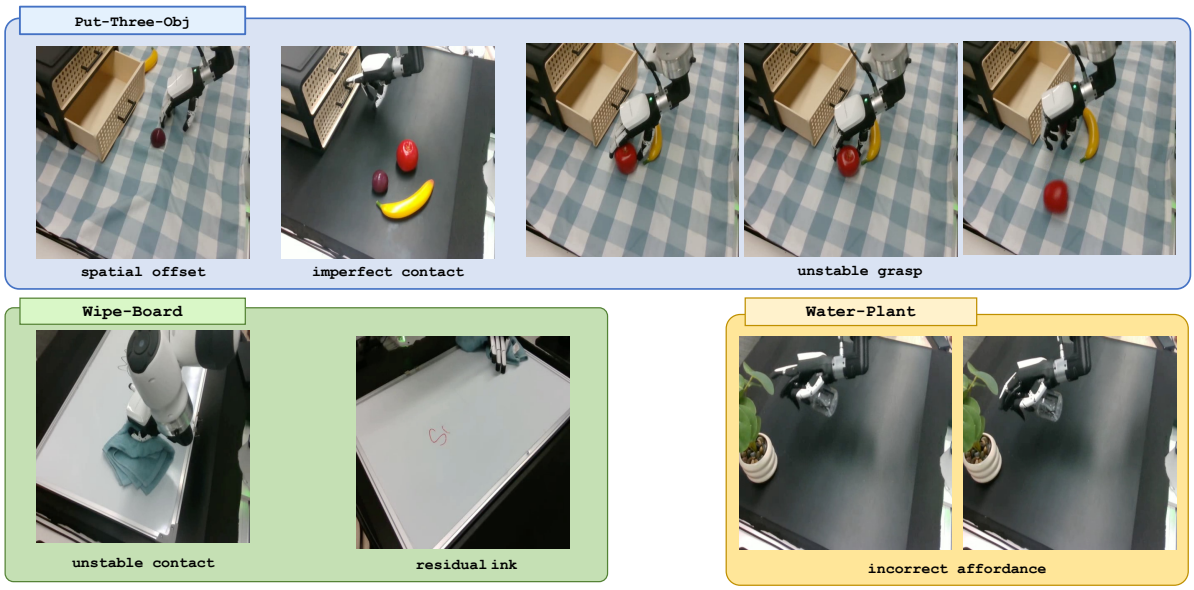

Figure 11. **Failure cases across real-robot tasks**. `Put-Three-Obj`: spatial misalignment, incomplete contact, and unstable grasp. `Wipe-Board`: insufficient planar contact and persistent residual ink. `Water-Plant`: incorrect affordance reasoning leading to wrong bottle orientation. These cases highlight the remaining challenges in precise alignment, contact stability, and fine-grained affordance modeling.

D.5. Analysis of Failure Cases

To complement the successful rollouts discussed above, we further analyze representative failure modes observed in the real-robot evaluation. These cases, visualized in Figure 11, highlights the remaining challenges faced by JALA in precise alignment, contact stability, and affordance understanding.

Put-Three-Obj. Several failures arise from subtle but compounding perception–control mismatches. First, *spatial offset* occurs when the predicted pre-grasp pose is misaligned by a few centimeters, leading the fingers to contact the fruit at an unintended angle. Second, *imperfect contact* is frequently observed when the Inspire hand touches the object but does not achieve a stable grasp envelope; in such cases, the object may roll or rotate, preventing a clean lift. Finally, *unstable grasp* can occur even after initial contact is made: due to small wrist-orientation errors or insufficient finger closure, the fruit slips during transport, especially the elongated banana whose geometry amplifies torque instability. These patterns indicate that although JALA learns strong recovery behaviors in many successful trials, its grasping robustness still degrades under challenging object geometries and unseen textures.

Wipe-Board. Two primary failure factors are observed. First, *unstable contact* arises when the cloth is not pressed firmly or flatly enough against the board surface, leading to ineffective wiping motion. Minor deviations in wrist roll or pitch cause the cloth to fold or lose planar contact, reducing cleaning efficacy. Second, *residual ink* remains when the model fails to revisit lightly marked regions, suggesting that the closed-loop perception sometimes underestimates faint ink traces or overestimates wiping success. Compared with the adaptive multi-pass behavior seen in successful rollouts, these weaker cases reveal that visual sensitivity to subtle color contrast changes is still a limiting factor.

Water-Plant. Most failures arise from *incorrect affordance* reasoning. In these cases, the policy grasps the spray bottle, but positions it at an incorrect orientation relative to the plant or the trigger mechanism. As shown in Figure 11, the model occasionally brings the bottle close to the plant but does not align the nozzle toward the target, or fails to align the index-finger actuation direction with the trigger. This indicates that while global positioning is largely reliable, precise end-effector orientation and fine-grained affordance understanding remain areas for further improvement.