

Touch What You Say: Contact-Local Tactile Grounding for Physical-Property Instructions

Le Wang, Kelan Peng, Chenyu Zhang, Wenjuan Han*

Key Laboratory of Big Data & Artificial Intelligence in Transportation

(Beijing Jiaotong University), Ministry of Education

School of Computer Science and Technology, Beijing Jiaotong University, Beijing 100044, China

Abstract

While vision-language-action (VLA) models excel at robotic manipulation, they frequently fall short when interpreting physical-property language, where adjectives such as *fragile* or *coarse-grained* specify constraints that are often visually ambiguous and should modulate behavior primarily during active contact. We therefore propose Contact-Local Semantic Actuation (CLSA), an architecture that extracts semantically structured tactile representations through a language-guided bottleneck and injects them through a contact-gated residual path, confining physical adjustments to contact-relevant stages without perturbing stable nominal motion. To probe instruction-conditioned behavior under matched contexts, we introduce the Action-Property-Constraint (APC) hierarchy. On the language side, APC contrasts action-only (L0), property-aware (L1), and constraint-explicit (L2) prompts; on the trajectory side, offline evaluation uses the corresponding demonstrated behavior strata defined by the data protocol. Across matched offline comparisons, a 140-trial comparative real-robot benchmark, and a separate 240-rollout APC sweep on four tactile-sensitive tasks, CLSA improves instruction sensitivity while maintaining action accuracy. We also report visually ambiguous black-bag trials within the comparative benchmark as a diagnostic of touch-based material recognition after contact. Code is available at <https://github.com/melodyiii/CLSA>.

1 Introduction

Vision-language-action (VLA) models (Brohan et al., 2023; Kim et al., 2024; Octo Model Team et al., 2024) have made substantial progress in mapping task instructions to robot actions. By coupling pretrained vision-language representations

with robot policy learning, they improve semantic generalization across tasks and objects. Yet their strength remains primarily visual and semantic: under visual occlusion or ambiguity, vision alone often cannot determine how control should be modulated once physical interaction begins. This limitation has motivated recent research to integrate tactile signals into VLA pipelines.

While recent tactile-augmented VLAs have explored token-level fusion (Hao et al., 2025) or feature conditioning and modulation (Bi et al., 2025; Morissette et al., 2026), they frequently expose touch through a globally shared policy context. This raises two coupled issues. First, global injection can perturb the action distribution across all movement phases, including pre-contact stages where tactile input adds sensor noise instead of task-relevant signal. Second, although recent efforts have begun to bridge vision, language, and touch (Huang et al., 2025), they do not explicitly specify which tactile cues physical-property words such as “fragile,” “soft,” or “coarse-grained” should select. As a result, grounding physical-property language into closed-loop manipulation remains an open routing problem: the policy must determine both *which* tactile evidence a descriptor refers to and *when* that evidence should be allowed to alter control.

To bridge this gap, we propose Contact-Local Semantic Actuation (CLSA), a dual-path architecture that separates stable nominal control from contact-phase semantic modulation. CLSA preserves a visuo-linguistic base path for macro-motion, while a language-guided cross-modal bottleneck with tactile-language alignment (TLA) extracts tactile semantics and injects them through a contact-gated residual path. As shown in Figure 1, once contact begins, the policy can adjust grip force for a fragile blueberry or distinguish visually identical bags by touch, without perturbing the approach motion.

*Corresponding author.

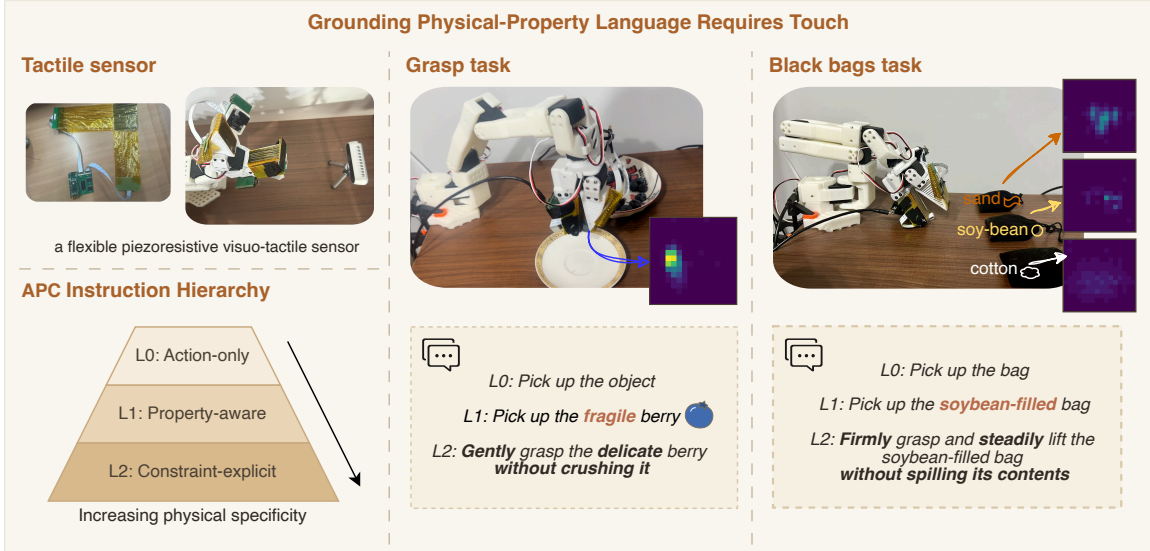


Figure 1: Bilateral fingertip tactile sensing, the APC instruction hierarchy, and representative real-robot tasks. Instructions are shown at three levels: L0, L1, and L2.

Existing evaluations mostly rely on binary task success, which fails to isolate how language conditions physical motion. To address this, we introduce the Action-Property-Constraint (APC) hierarchy to probe instruction-conditioned behavior under matched contexts. On the language side, APC ranges from action-only (L0) to richer property- and constraint-aware prompts; on the trajectory side, offline evaluation uses the matched demonstrated behavior strata defined by the data protocol. The APC gradient $\Delta(L2, L0)$ then tests whether richer physical-property language shifts predictions toward more-constrained demonstrated behavior under matched task and contact contexts. We pair it with Swap L2, which changes only the property descriptor on held-out L2 episodes as a cleaner prompt-side probe of adjective grounding.

We evaluate CLSA in two real-robot settings. The first is a 140-trial comparative benchmark across three groups: fragile grasping (blueberry, egg), compliance-sensitive manipulation (cloth, sponge), and visually ambiguous sorting (the black-bag family). The second is a separate APC sweep comprising 240 CLSA rollouts on four tactile-sensitive tasks and measuring how peak force and success change across L0, L1, and L2 prompts. Because bag contents are visually matched, the black-bag tasks serve as a targeted diagnostic for whether post-contact tactile signals can support material-specific action selection. In offline evaluation, CLSA achieves negative Δ gradients across all evaluated tasks under

this matched protocol, with notable improvements on compliance and black-bag settings. In brief, our results show that contact-local semantic routing enforces physical constraints without sacrificing base control.

Our primary contributions are:

- **Architecture:** We propose CLSA, a mechanism that links property-centric instructions to physical interaction by extracting semantically structured tactile representations and releasing them through a contact-gated residual path.
- **Formulation:** We introduce the APC hierarchy and the corresponding $\Delta(L2, L0)$ metric to probe instruction-conditioned behavior under matched task and contact contexts.
- **Empirical validation:** Through controlled ablations and real-robot experiments, we show that CLSA outperforms unstructured tactile fusion baselines, particularly on visually ambiguous tasks where physical properties must be decoded from touch.

2 Related Work

Vision-Language-Action Models. Recent VLA and generalist robot policies map visual observations and language instructions to robot actions, including RT-1/RT-2 (Brohan et al., 2022, 2023), OpenVLA (Kim et al., 2024), Octo (Octo Model Team et al., 2024), and PaLM-E (Driess

et al., 2023). These systems combine language-conditioned perception with large-scale robot policy learning, and action-generation methods such as Diffusion Policy further show that expressive continuous action distributions improve visuomotor imitation (Chi et al., 2023). Their main strength lies in semantic task grounding and visual generalization. They do not explicitly address how physical-property words should select tactile evidence, or when such evidence should be allowed to modulate contact-level control.

Tactile and Visuo-Tactile Learning. Tactile sensing provides contact-centric evidence for grasping, slip detection, and material-aware manipulation (Dahiya et al., 2010; Calandra et al., 2018). Recent visuo-tactile policies show that touch improves manipulation when visual observations are incomplete or physically ambiguous, especially in fine-grained and contact-rich settings (Huang et al., 2024). This line of work establishes that tactile feedback improves robust control, but it largely treats touch as a perceptual or control signal rather than asking how language selects tactile cues relevant to a physical constraint.

Tactile-Augmented VLA and Multimodal Fusion. A growing set of methods augments VLA policies with tactile inputs, including TLA (Hao et al., 2025), VLA-Touch (Bi et al., 2025), Tactile-VLA (Huang et al., 2025), and TacFiLM (Morissette et al., 2026). These works bridge vision, language, touch, and action through token-level fusion, feature conditioning, modulation, or tactile feedback, and demonstrate the benefit of tactile signals for language-conditioned manipulation. CLSA is complementary but focuses on a different question: how physical-property language selects tactile evidence and routes it into the policy only when contact makes the referenced property actionable.

3 Method

CLSA treats physical-property grounding as two routing problems: selecting the tactile evidence named by the instruction and using it only when contact makes it actionable for control. It combines an APC-conditioned flow-matching policy with a language-guided semantic tactile state and a contact-gated residual path trained in two stages.

3.1 Problem Setup and Contact-Local Design

We model language-conditioned manipulation as continuous action generation. At time step t , the policy receives $O_t = \{I_t, g_t, q_t\}$, where $I_t = \{I_t^{\text{front}}, I_t^{\text{side}}\}$ contains synchronized front-view and side-view RGB images, $g_t \in \mathbb{R}^{H \times 2 \times 16 \times 16}$ is the bilateral tactile history, and $q_t \in \mathbb{R}^{d_q}$ is the robot proprioceptive state. The instruction is indexed by an APC level $\ell \in \{L0, L1, L2\}$: L0 gives action-only guidance, L1 names a physical property, and L2 adds an explicit contact-dependent constraint such as “grasp the fragile object gently.”

The policy predicts actions with conditional flow matching (Lipman et al., 2022). For a noisy intermediate action $\tilde{a}_t^\rho$ at flow time $\rho \in [0, 1]$, the model estimates a velocity field $v_\theta(\tilde{a}_t^\rho, \rho)$ that transports a simple noise distribution toward the target action trajectory. Physical-property language adds a routing decision to this objective: the policy must determine which tactile evidence the instruction refers to and when that evidence should change the velocity field.

The contact-local design follows from the temporal structure of tactile evidence. Before contact, tactile history mostly reflects no interaction or sensor noise; during contact, it carries the physically meaningful cues that should modulate behavior. CLSA captures this asymmetry with two action pathways: a base path for nominal execution across the full trajectory and a semantic path that predicts an instruction-conditioned residual released by contact evidence. Figure 2 summarizes this structure: language and vision guide a tactile bottleneck, which yields a semantic tactile state whose effect on action generation is controlled by a contact gate.

3.2 Language-Guided Tactile Semantic Bottleneck

The bottleneck addresses the first part of routing: which tactile evidence the instruction should read. A single tactile history can support different physical interpretations under different property words, so the semantic state must be conditioned on language. We encode the instruction $x_{1:N}^{(\ell)}$ and the dual-view RGB pair I_t with a Qwen2-VL backbone (Wang et al., 2024), producing a joint vision-language sequence $H_t \in \mathbb{R}^{N' \times d_v}$ and an instruction summary $h_t \in \mathbb{R}^{d_v}$. A proprioceptive projection of q_t is fused with the policy features used by the base controller, while a bilateral CNN-GRU

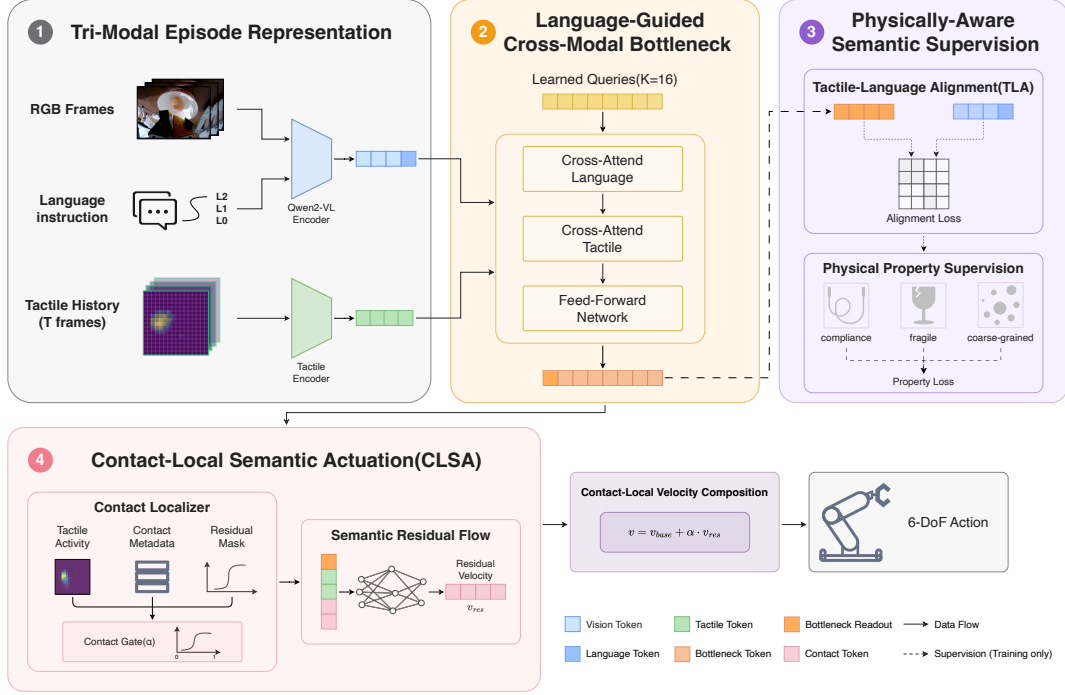


Figure 2: Overview of CLSA. RGB observations, tactile history, and the instruction are encoded into a language-guided cross-modal bottleneck, which is structured by tactile-language alignment and physical-property supervision. The resulting semantic state enters the policy through a contact-local residual action path, while the base path remains responsible for stable control across the full trajectory.

tactile encoder maps the tactile history g_t to a compact tactile state $z_t \in \mathbb{R}^{d_t}$ (Cho et al., 2014; Huang et al., 2024).

The language-guided cross-modal bottleneck (LGB) uses K learned queries to map z_t to an instruction-conditioned tactile representation. At layer r , the queries first read the projected vision-language sequence and then use the updated query state to read the projected tactile token:

$$\begin{aligned}\bar{Q}^r &= Q^{r-1} + \text{CA}(Q^{r-1}, L_t, L_t), \\ Q^r &= \bar{Q}^r + \text{CA}(\bar{Q}^r, Z_t, Z_t),\end{aligned}\quad (1)$$

where $Q^r \in \mathbb{R}^{K \times d_b}$ is the query matrix, $L_t = H_t W_H \in \mathbb{R}^{N' \times d_b}$ is the projected vision-language sequence, $Z_t = z_t W_Z \in \mathbb{R}^{1 \times d_b}$ is the projected tactile token, and $\text{CA}(Q, K, V)$ denotes multi-head cross-attention (Vaswani et al., 2017). The first attention step binds each query to the instruction context. The second step reads tactile evidence through that language-conditioned query state.

The final bottleneck’s first token is projected to the semantic tactile state $z_t^{\text{sem}} \in \mathbb{R}^{d_b}$. Tactile-Language Alignment (TLA) structures this state by matching semantic tactile states to instruction summaries through physical-property overlap. For

samples i and j , the target $s_{ij} \in [0, 1]$ comes from task-level property labels such as compliance, fragility, and granular structure; Appendix F gives the full construction:

$$\mathcal{L}_{\text{TLA}} = \frac{1}{B^2} \sum_{i,j} \left(\sigma \left(\frac{e_i^\top l_j}{\tau} \right) - s_{ij} \right)^2, \quad (2)$$

where B is the batch size, e_i and l_j are L_2 -normalized projections of z_i^{sem} and h_j , and τ is a learnable temperature. An auxiliary physical-property classifier with loss $\mathcal{L}_{\text{phys}}$ is also applied to z_t^{sem} , which is then used as the tactile input to the semantic residual path.

3.3 Contact-Local Actuation and Training

The actuation module addresses the second part of routing: when the semantic tactile state may change the action distribution. The raw tactile state z_t provides contact activity and local tactile dynamics. The semantic state z_t^{sem} provides the property-conditioned evidence extracted by the bottleneck. The raw tactile state and proprioceptive state q_t enter the base controller. The semantic state conditions the residual branch that predicts the velocity correction.

Contact prior. We compute a scalar contact prior $m_t \in [0, 1]$ for the current frame. During training and offline evaluation, when valid post-hoc contact-timing annotations are available, they are blended with an online tactile activity score to provide a phase-timing scaffold for semantic modulation. During real-robot deployment, or whenever annotations fail the validity checks, the same prior is computed from the online score alone because post-hoc timing annotations are unavailable. The online score uses the maximum taxel activation and the mean contact indicator over a sliding 16-frame tactile window; Appendix G gives the full construction. The prior controls only the timing of semantic modulation. Property identity and correction direction are supplied by the language-conditioned representation and the residual path.

Semantic gate. A lightweight MLP predicts the residual confidence conditioned on semantic tactile evidence, raw tactile state, and contact prior:

$$\alpha_t = m_t \cdot \text{gate}_\theta(z_t^{\text{sem}}, z_t, m_t), \quad (3)$$

where $\alpha_t \in [0, 1]$ is the final residual gate. The multiplicative factor m_t provides an explicit phase prior. The MLP uses the same prior as an input feature, allowing residual confidence to depend on both contact strength and the current tactile-language state.

Velocity composition. The flow-matching velocity is composed from a base path and the gated semantic residual:

$$v_\theta(\tilde{a}_t^\rho, \rho) = v_{\text{base}}(\tilde{a}_t^\rho, \rho, h_t, z_t, q_t) + \alpha_t v_{\text{sem}}(\tilde{a}_t^\rho, \rho, z_t^{\text{sem}}, z_t, m_t). \quad (4)$$

The base velocity v_{base} handles nominal macro-motion using the instruction summary, raw tactile state, and proprioceptive state. The semantic velocity v_{sem} predicts property-specific corrections from z_t^{sem} , with z_t and m_t providing contact context. This composition preserves a full-trajectory control path and gives physical-property language a separate contact-gated route into action.

Training follows the same split. **Stage A** trains the semantic bottleneck with $\mathcal{L}_A = \lambda_{\text{tla}}\mathcal{L}_{\text{TLA}} + \lambda_{\text{phys}}\mathcal{L}_{\text{phys}}$, giving z_t^{sem} a property-aligned geometry before it drives residual control. **Stage B** optimizes flow matching while retaining the alignment and property losses, together with a non-contact suppression term $\mathcal{L}_{\text{NC}}$ weighted by $(1 - m_t)^2$ and a residual regularizer $\mathcal{L}_{\text{res}}$.

To enforce stronger residual use under stronger physical-property instructions, we add an APC monotonicity loss over matched task-contact groups. For each group g containing L0, L1, and L2 samples, we require the mean gated residual norm $\bar{r}_{g,\ell} = \mathbb{E}[\|\alpha_t v_{\text{sem}}\|_2]$ to increase with APC level:

$$\mathcal{L}_{\text{mono}} = \frac{1}{|G|} \sum_{g \in G} \left(\max(0, \gamma - (\bar{r}_{g,\text{L1}} - \bar{r}_{g,\text{L0}})) + \max(0, \gamma - (\bar{r}_{g,\text{L2}} - \bar{r}_{g,\text{L1}})) \right), \quad (5)$$

where γ is a margin. This encourages stronger instructions to recruit more semantic residual correction under matched contexts.

The full Stage B objective is $\mathcal{L}_B = \mathcal{L}_{\text{FM}} + \lambda_{\text{tla}}\mathcal{L}_{\text{TLA}} + \lambda_{\text{phys}}\mathcal{L}_{\text{phys}} + \lambda_{\text{nc}}\mathcal{L}_{\text{NC}} + \lambda_{\text{res}}\mathcal{L}_{\text{res}} + \lambda_{\text{mono}}\mathcal{L}_{\text{mono}}$.

Stage A defines the property-structured tactile state. Stage B learns the base velocity, the semantic residual, and the contact gate that controls how the semantic state enters continuous action generation.

4 Experiments

We structure the evaluation around three questions.

RQ1. Does CLSA increase instruction sensitivity while preserving control fidelity? **RQ2.** Do its gains depend on the semantic bottleneck, tactile-language alignment, and contact-local residual route? **RQ3.** Does the learned modulation depend on coherent tactile evidence? The three are linked: the main comparison addresses RQ1, ablations test RQ2, and mechanism probes address RQ3.

4.1 Evaluation Metrics

Our evaluation combines control accuracy, instruction sensitivity, and real-robot success. We report normalized action MSE and define the **APC gradient** $\Delta(\ell_2, \ell_1) = \mathcal{E}_{\ell_2} - \mathcal{E}_{\ell_1}$, where $\mathcal{E}_\ell$ is the mean squared action error under APC level ℓ . A negative $\Delta(\text{L2}, \text{L0})$ indicates closer alignment to more-constrained demonstrated behavior strata under L2 than under L0 within matched task and contact contexts. We also use adjective-swap minimal pairs (Swap L2), which change only the property descriptor on fixed held-out L2 episodes and therefore serve as a closer prompt-side probe of adjective-level grounding.

Method	Semantic routing	MSE ↓	$\Delta(L2, L0)$ ↓	Swap ↑	Neg. ↑
<i>Baselines and references</i>					
No Tactile	none	3.79	+0.002	1.35	3/7
Tactile Concat	late feature concat	3.98	−0.063	1.40	5/7
TLA + Global Residual	non-local residual correction	4.40	−0.087	2.22	6/7
pi0.5 [†]	external RGB-only VLA	3.48	−0.076	1.72	4/7
CLSA (Ours)	contact-local residual velocity field	3.64	−0.092	2.70	7/7

Table 1: Offline comparison on seven held-out tasks. No Tactile and Tactile Concat are matched baselines, TLA + Global Residual is a matched VTLa-family reference, and pi0.5 is shown as an external reference. CLSA is listed last for direct comparison. [†]pi0.5 uses a different action space and tokenizer, so its APC-gradient, Swap L2, and negative- Δ metrics are approximate. All results are averaged over 3 random seeds.

4.2 Experimental Setup

We use a LeRobot SO-101 arm (Cadene et al., 2026) with bilateral 16×16 piezoresistive tactile arrays and synchronized front-view and side-view RGB cameras. The dataset contains 228 episodes across seven tasks, collected under L0 and L2 instructions and post-hoc relabeled into APC levels from tactile-force signatures; L1 denotes intermediate demonstrations between the L0 and L2 contact regimes (Appendix E).

The tasks cover fragile grasping, compliance-sensitive manipulation, and visually matched black-bag sorting. For black-bag real-robot evaluation only, all methods use a shared deployment-time material readout from post-contact tactile windows; it is not used in offline evaluation or Stage B training.

Real-robot evaluation includes a 140-trial comparative benchmark over all seven tasks (Table 4) and a 240-rollout APC sweep on four tactile-sensitive tasks (Table 2).

We compare the No Tactile baseline, the Tactile Concatenation baseline with late feature fusion, and pi0.5 (Physical Intelligence et al., 2025) as an external reference. We also report TLA + Global Residual as a matched VTLa-family routing reference. Metrics include normalized action MSE, $\Delta(L2, L0)$, Swap L2, contact-phase MSE, and real-robot success rate.

4.3 Main Results

To answer RQ1, Table 1 compares CLSA against matched baselines and shows that, among matched VTLa-family models, it gives the best combination of control fidelity and instruction sensitivity, with the strongest APC gradient, Swap, and negative- Δ coverage and lower MSE than No Tactile, Tactile Concatenation, and TLA + Global

Task	Peak force mean ↓			Success rate ↑		
	L0	L1	L2	L0	L1	L2
blueberry grasp	218.6	157.4	113.1	13/20	15/20	19/20
egg grasp	203.5	138.2	96.7	12/20	16/20	20/20
gentle sponge	213.3	112.0	103.4	5/20	6/20	14/20
inboxpicking-cloth	251.1	152.5	112.5	3/20	6/20	9/20

Table 2: APC-level peak force and success rate from a separate 240-rollout CLSA-only real-robot sweep on four tactile-sensitive tasks. Lower force and higher success under L2 indicate gentler, more reliable property-conditioned execution.

Method	Contact MSE ↓
No Tactile	3.82
Tactile Concat	3.85
pi0.5 [†]	3.34
CLSA (Ours)	3.04

Table 3: Contact-phase action error on contact frames only, normalized on the same scale as Table 1. [†]pi0.5 remains an external reference because it uses a different action space and tokenizer.

Residual. pi0.5 attains lower aggregate MSE but remains an external reference because it uses a different action space and tokenizer. The baseline pattern supports the routing claim: No Tactile shows little instruction sensitivity, Tactile Concatenation improves coarse APC separation without robust adjective grounding, and TLA + Global Residual incurs a control cost from non-local semantic injection.

Table 2 shows that richer APC instructions consistently reduce peak force and improve success on tactile-sensitive real-robot tasks. We observe the same L0-to-L2 trend across all four tasks in a separate CLSA-only APC sweep, indicating gentler and more reliable execution under constraint-explicit instructions.

Contact-phase error tests the routing hypothesis

at the frames where tactile evidence can affect control. Table 3 shows that CLSA reduces contact-phase error by 20% over No Tactile and 21% over Tactile Concatenation. Its error is also lower than the $\pi 0.5$ reference (3.04 vs. 3.34) and lower than its own aggregate error (3.04 vs. 3.64), indicating that the tactile path improves the contact phase without reducing full-trajectory control fidelity.

Real-robot trials preserve the offline trend under execution noise. Table 4 first summarizes the standalone-policy comparison on the fragile and compliance tasks, where CLSA reaches 62/80 overall successes, compared with 34/80 for No Tactile, 39/80 for Tactile Concatenation, and 43/80 for $\pi 0.5$. The black-bag rows then report the shared black-bag deployment evaluation described in Section 4.2; with these rows included, the system-level total is 89/140. Fragile grasping is near saturation, so the largest margins appear in compliance and black-bag tasks. These tasks underdetermine the named property from vision and require post-contact tactile evidence for material-consistent actions. Black-bag success rates therefore measure performance under this shared deployment-time readout on a visually ambiguous tactile diagnostic.

Remaining failures concentrate in black-bag sorting and inboxes-picking-cloth, consistent with the limits of coarse 16×16 tactile resolution and high-variance deformable-object grasps. Gains are largest in these difficult settings, where visual evidence provides the weakest constraint on the required physical behavior.

Taken together, the offline, contact-phase, and real-robot results answer RQ1 affirmatively: CLSA increases instruction sensitivity while preserving control fidelity, especially where tactile evidence is most decisive.

4.4 Ablation Study

To answer RQ2, we ablate the components that support instruction-sensitive contact control: the language-guided gate, the tactile-language alignment objective, and the contact-local residual route.

The ablations in Table 5 show that CLSA requires both semantic alignment and contact-local routing. Fixed-LGB preserves a strong APC gradient but collapses Swap from 2.70 to 0.92, indicating coarse instruction-level sensitivity without reliable adjective-level discrimination. Removing TLA weakens MSE and negative- Δ coverage, sug-

gesting that alignment structures the bottleneck along reusable physical-property axes.

The TLA + Global Residual variant isolates the routing effect. It uses the same language-guided bottleneck and alignment as CLSA, but injects the semantic state globally. It recovers part of the instruction effect, confirming that the bottleneck carries useful physical information. Its MSE remains substantially worse than CLSA’s (4.40 vs. 3.64), showing the control cost of non-local semantic injection. Performance depends on how tactile semantics enter control, not only on whether they are present.

These ablations answer RQ2: the gains depend jointly on the semantic bottleneck, tactile-language alignment, and contact-local residual route.

4.5 Mechanism Evidence

RQ3 asks whether CLSA grounds property words in tactile evidence. The mechanism probes test this directly. Figure 3(a) holds a contact frame fixed and varies only the APC level. Attention shifts from generic task wording under L0 toward property-bearing tokens such as *fragile* and *breakable* under L2, indicating that the model reads tactile input through the named property.

This qualitative pattern matches the Stage A ablation. With TLA, the mean cosine distance between bottleneck embeddings from different physical-property groups is 0.677; removing TLA collapses the distance to 0.018 (Table 7, Appendix B). Thus, the bottleneck is not only language-sensitive; it is organized along physical-property axes.

The learned bottleneck geometry shows the same property structure. In Figure 3(b), the t-SNE view (van der Maaten and Hinton, 2008) separates fragile, fragile+compliant, and compliant groups into distinct regions, consistent with the large cross-property cosine gap preserved by TLA.

Residual separation requires coherent tactile structure. With intact tactile input, semantic residuals remain separated in direction and magnitude, but temporal shuffling shrinks the mean cosine and L_2 distances from 0.41/0.31 to 0.15/0.11, and zero masking reduces them further to 0.06/0.04. Once post-contact structure is destroyed, the separation disappears, showing that adjective-conditioned modulation relies on tactile evidence rather than prompt statistics alone.

These probes answer RQ3: the learned modu-

Group	Task	No Tactile	Tactile Concat	pi0.5 [†]	CLSA
Fragile	blueberry grasp	12/20	13/20	16/20	19/20
	egg grasp	13/20	14/20	15/20	20/20
	subtotal	25/40	27/40	31/40	39/40
Compliance	inboxpicking-cloth	4/20	3/20	5/20	9/20
	gentle sponge	5/20	9/20	7/20	14/20
	subtotal	9/40	12/40	12/40	23/40
Core overall	fragile + compliance	34/80	39/80	43/80	62/80
Black-bag	cotton bag	3/20	6/20	4/20	8/20
	sand bag	5/20	8/20	5/20	10/20
	soybeans bag	5/20	7/20	5/20	9/20
	subtotal	13/60	21/60	14/60	27/60
System overall	all tasks	47/140	60/140	57/140	89/140

Table 4: Real-robot successes on the 140-trial comparative benchmark. Each task has 20 trials. Core overall summarizes the fragile and compliance tasks (80 trials). Black-bag rows and System overall use the shared deployment-time material readout in Section 4.2. [†]pi0.5 is an external reference.

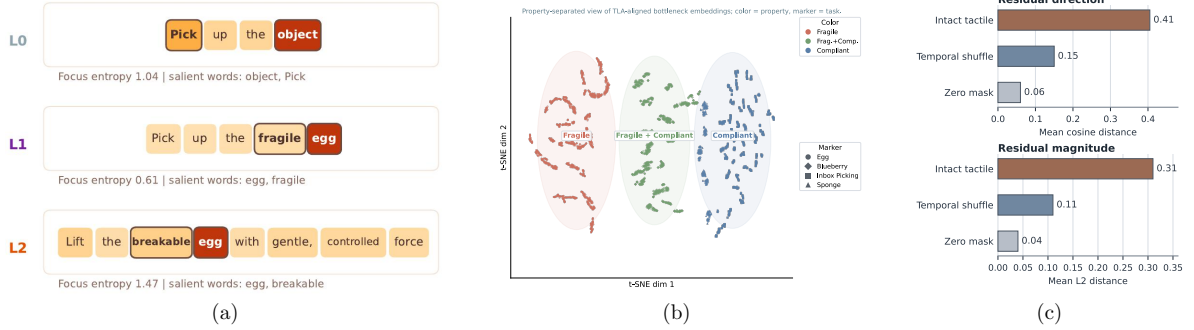


Figure 3: Mechanism evidence. (a) Attention sharpens from L0 to L2. (b) Stage A bottleneck embeddings cluster by physical property. (c) Residual separation collapses under shuffled or zeroed tactile input.

Variant	MSE ↓	$\Delta(L2, L0)$ ↓	Swap ↑	Neg. ↑
Full CLSA	3.64	-0.092	2.70	7/7
Fixed-LGB	4.77	-0.096	0.92	5/7
No TLA	4.72	-0.088	2.31	5/7
TLA + Global Residual	4.40	-0.087	2.22	6/7

Table 5: Controlled ablations. All variants share the same backbone, data, and evaluation harness.

lation depends on coherent tactile evidence rather than prompt statistics alone.

5 Conclusion

This paper studies whether physical-property language can shape manipulation when the decisive evidence emerges only at contact. Across APC-based evaluation, adjective-swap probes, and real-robot trials, the results support a simple claim: touch helps only when the policy reads tactile evidence through the named property and releases that information at contact. Touch is just contact data until language gives it structure. Language is still incomplete until contact provides ev-

idence the action can use. For multimodal agents, the key question is not just which modalities are present, but whether language, perception, and control meet when contact starts to matter.

6 Limitations

Our results have several limits: the SO-101 platform lacks high-precision force control, the 16×16 tactile arrays may miss fine material cues, and current CLSA training and offline analysis still use a hybrid contact prior, with post-hoc timing when annotations are available. Real-robot deployment uses only the online tactile activity score, so the black-bag benchmark is best read as a two-stage deployment diagnostic rather than a pure per-timestep test of continuous closed-loop grounding.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62406020), the Fundamental Research Funds for the Central Universities under Grant No. 2025JBZX058,

the CHN Railway Grant (No. RJ2025Z001), and Beijing Urban Construction Group Co., Ltd. (No. K24L01470).

References

- Jianxin Bi, Kevin Yuchen Ma, Ce Hao, Mike Zheng Shou, and Harold Soh. 2025. VLA-Touch: Enhancing vision-language-action models with dual-level tactile feedback. *arXiv preprint arXiv:2507.17294*.
- Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, and 35 others. 2023. RT-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*.
- Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, and 32 others. 2022. RT-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*.
- Remi Cadene, Simon Alibert, Francesco Capuano, Michel Aractingi, Adil Zouitine, Pepijn Kooijmans, Jade Choghari, Martino Russi, Caroline Pascal, Steven Palma, Mustafa Shukor, Jess Moss, Alexander Soare, Dana Aubakirova, Quentin Lhoest, Quentin Gallouédec, and Thomas Wolf. 2026. **LeRobot: An open-source library for end-to-end robot learning**. *arXiv preprint arXiv:2602.22818*.
- Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H. Adelson, and Sergey Levine. 2018. More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and Automation Letters*, 3(4):3300–3307.
- Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. 2023. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*.
- Kyunghyun Cho, Bart van Merriënboer, Çağlar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.
- Ravinder S. Dahiya, Giorgio Metta, Maurizio Valle, and Giulio Sandini. 2010. Tactile sensing – from humans to humanoids. *IEEE Transactions on Robotics*, 26(1):1–20.
- Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Ser-manet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, and 3 others. 2023. PaLM-E: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*.
- Peng Hao, Chaofan Zhang, Dingzhe Li, Xiaoge Cao, Xiaoshuai Hao, Shaowei Cui, and Shuo Wang. 2025. TLA: Tactile-language-action model for contact-rich manipulation. *arXiv preprint arXiv:2503.08548*.
- Binghao Huang, Yixuan Wang, Xinyi Yang, Yiyue Luo, and Yunzhu Li. 2024. 3D-ViTac: Learning fine-grained manipulation with visuo-tactile sensing. In *Conference on Robot Learning*. ArXiv:2410.24091.
- Jialei Huang, Shuo Wang, Fanqi Lin, Yihang Hu, Chuan Wen, and Yang Gao. 2025. Tactile-VLA: Unlocking vision-language-action model’s physical knowledge for tactile generalization. *arXiv preprint arXiv:2507.09160*.
- Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. 2024. OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*.
- Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.
- Charlotte Morissette, Amin Abyaneh, Wei-Di Chang, Anas Houssaini, David Meger, Hsiu-Chin Lin, Jonathan Tremblay, and Gregory Dudek. 2026. Tactile modality fusion for vision-language-action models. *arXiv preprint arXiv:2603.14604*.
- Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. 2024. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*.
- Physical Intelligence. 2024. OpenPI. <https://github.com/Physical-Intelligence/openpi>. GitHub repository.
- Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Nicolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, and 17 others. 2025. $\pi_{0.5}$: a vision-language-action model with open-world generalization.

- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763.
- Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-VL: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

A Per-Task APC Gradient Breakdown

Table 6 reports the per-task APC gradient $\Delta(L2, L0)$ for CLSA on the normalized MSE scale. All seven tasks yield negative Δ , indicating closer alignment to constrained L2 demonstrations than to natural-motion L0 demonstrations within the corresponding APC behavior strata. The gradients are small on visually informative tasks and larger on the black-bag family, where vision gives weaker evidence about the physical property. The final column lists each task’s primary physical-property axis.

Task	$\Delta(L2, L0)$	PPA axis
inboxpicking-cloth	−0.013	compliance
blueberry grasp	−0.017	fragility
egg grasp	−0.019	extreme fragility
gentle sponge	−0.038	compliance discrim.
cotton bag	−0.214	compliance (occluded)
sand bag	−0.184	fine-grained support
soybeans bag	−0.154	coarse-grained
All tasks (aggregate)	−0.092	—

Table 6: Per-task APC gradient for CLSA, normalized by $\sigma_{V_{TLA}}^2 = 808.1$. Negative values indicate closer alignment to constrained L2 demonstrations than to natural-motion L0 demonstrations across the corresponding APC behavior strata. The aggregate row is the frame-weighted mean reported in Table 1.

B Stage A Bottleneck Ablation

Stage A trains the semantic bottleneck before action learning, using the tactile-language alignment loss and the auxiliary physical-property classifier. Table 7 isolates the contribution of each objective on the held-out sweep. The representation-best checkpoint maximizes alignment score; the training-best checkpoint minimizes training loss.

Variant	Repr-best		Train-best	
	Align↑	CosGap↑	Align↑	CosGap↑
Formal Control	0.659	0.677	0.657	0.672
No TLA	0.505	0.018	0.505	0.015
No PhysProp	0.672	0.732	0.672	0.732

Table 7: Stage A ablation. Align = tactile-language alignment score; CosGap = mean cosine distance between bottleneck embeddings of different physical-property groups. PhysProp accuracy is 1.00 for Formal Control and No TLA, and 0.586 for No PhysProp. All metrics are computed on the same held-out sweep.

TLA controls the geometry of the bottleneck space. Removing TLA collapses the cosine gap

from 0.677 to 0.018 and lowers alignment from 0.659 to 0.505, showing that the alignment objective separates representations by physical property.

The auxiliary classifier preserves class-discriminative property information. Removing it keeps the alignment score high (0.672 vs. 0.659) and increases the cosine gap slightly (0.732 vs. 0.677), but classifier accuracy falls from 1.00 to 0.586. This failure propagates into the Stage B ablations in Table 5, where missing property supervision weakens action performance.

C Real-Robot Deployment Details

Real-world deployment is conducted on an SO-101 leader-follower platform. The follower arm serves as the execution platform, while the leader arm is used only for teleoperated demonstration collection during data acquisition. The deployed sensory interface consists of front-view RealSense RGB, side-view RGB, and bilateral tactile sensing. Specifically, an Intel RealSense D435i provides the front-view RGB stream and a side-view USB camera provides an auxiliary RGB stream; both cameras operate at 640×480 resolution and 30 Hz. In addition, two 16×16 tactile arrays are mounted on the left and right gripper fingers to provide contact-state and local force-distribution measurements. To preserve consistency between training and deployment, we reuse the same robot calibration, camera placement, and control frequency as in data collection, and maintain a 30 Hz real-time control loop during deployment.

At each control cycle, the system first acquires proprioceptive observations from the robot, including the positions of the six arm joints, together with RGB observations from the RealSense and side-view cameras. These raw observations are converted into the same input structure used during training: proprioceptive state q_t , dual-view RGB images I_t , task-level language conditioning, and a sliding tactile history g_t . Tactile sensing is acquired asynchronously: the left and right tactile arrays are continuously sampled at 50 Hz and stored in a sliding temporal window of 16 frames, yielding a temporally structured tactile observation aligned with the current control step. This design allows the deployed system to use visual, language, and tactile information jointly while preserving a stable control frequency.

During inference, the bilateral tactile window is

first encoded by the tactile branch, where convolutional layers extract spatial contact patterns and a GRU aggregates short-horizon tactile dynamics into a 512-dimensional tactile state. In the deployed CLSA policy, this tactile state is not appended as a generic tactile-token stream. Instead, it is combined with the language-conditioned visual representation through the language-guided bottleneck to form a semantic tactile state, which influences action only through the contact-local residual pathway when contact evidence is sufficiently strong. Because post-hoc contact-timing annotations are unavailable on hardware, the deployment-time contact prior is computed from the online tactile activity score alone. This design preserves the stability of the base control path during non-contact motion while allowing tactile semantics to modulate behavior once physical interaction becomes informative. After post-processing, the model output is converted into a 6-dimensional action command and sent to the follower arm. For safe execution on hardware, per-step action increments are clipped before being sent to the robot, and each episode is terminated after a fixed maximum number of control steps.

After each rollout, an operator annotates success or failure according to the predefined task criteria. In addition to task success, we log the number of executed control steps and average inference latency as supplementary deployment diagnostics. The main paper reports task success rate as the primary real-robot metric.

D APC Instruction Construction

The L0, L1, and L2 instruction templates for each task were generated using ChatGPT and manually verified for correctness and consistency. For each task, we provided the task description and the relevant physical property adjective (e.g., *fragile* for blueberry, *coarse-grained* for soybeans), and ChatGPT produced three instruction variants following the APC hierarchy: an action-only prompt (L0), a property-aware prompt (L1), and a constraint-explicit prompt (L2). All generated templates were reviewed by the authors to ensure that L1 and L2 differ only in the physical-property specification and that the constraint in L2 is executable by the robot platform. The resulting templates are included in the data release.

E Data Collection Protocol

For each task instance, we collected two physical demonstrations: one under an L0 (action-only) instruction and one under an L2 (constraint-explicit) instruction. Demonstrators followed the L2 constraint only during the L2 demonstration; L0 demonstrations were performed with natural, unconstrained motion.

After collection, episode-level L0/L1/L2 labels were assigned by an automated tactile classifier that extracts four contact-force features (peak taxel activation, mean contact force, maximum active-taxel ratio, and force gradient) from the 16×16 tactile arrays and assigns each episode to L0, L1, or L2 via a weighted voting scheme. Episodes with high contact forces are classified as L0, those with low forces as L2, and those in the intermediate range as L1. Task-specific thresholds are used for the black-bag family; for sponge tasks, hard/soft labels are set by folder-level ground truth. In this protocol, L1 denotes an intermediate behavior stratum defined by genuine teleoperated trajectories whose contact statistics fall in the intermediate range of the weak-label procedure. Every episode—regardless of its post-hoc APC level—retains its own physically demonstrated action trajectory. The L1 prompt text is derived from L2 templates by stripping the execution constraint while retaining the physical-property adjective, so the language side still instantiates the full APC hierarchy even though collection was performed under two instructed regimes.

Interpretation of APC levels. APC in this paper should be interpreted as a prompt-and-behavior protocol. The L0/L1/L2 templates define the language side of the hierarchy, while the episode-level weak labels define the corresponding behavior strata on the trajectory side. Accordingly, the offline APC split measures how stronger property language moves predictions toward more constrained demonstrated behavior strata under matched task and contact contexts while the APC prompt and behavior hierarchy are evaluated together. The APC gradient $\Delta(L2, L0)$ therefore compares alignment to more versus less constrained demonstrated behavior strata: L0 reflects unconstrained motion, L2 reflects constrained execution, and L1 captures naturally occurring intermediate-force behaviors.

Contact phases (approach, contact, lift, retreat) were automatically segmented from tactile signal

peaks in the 16×16 piezoresistive arrays and manually verified by the authors. A frame is labeled as contact when the maximum taxel activation exceeds a conservative threshold and the mean contact indicator over the 16-frame tactile window is above 0.5.

F TLA Loss Formulation

The tactile-language alignment loss matches the pairwise similarity between semantic tactile states and instruction summaries to physical-property overlap:

$$\mathcal{L}_{\text{TLA}} = \frac{1}{B^2} \sum_{i,j} \left(\sigma \left(\frac{e_i^\top l_j}{\tau} \right) - s_{ij} \right)^2, \quad (6)$$

where B is the minibatch size, $e_i \in \mathbb{R}^{256}$ is the normalized projection of z_i^{sem} , $l_j \in \mathbb{R}^{256}$ is the normalized projection of h_j , τ is a learnable temperature, and $s_{ij} \in [0, 1]$ is the pairwise physical-property similarity derived from weak labels for compliance, fragility, and granular structure. We use a sigmoid-plus-MSE objective because s_{ij} is a bounded overlap target, not a normalized distribution and not only a ranking signal. KL divergence would tie the loss to batchwise normalization, while a pure ranking loss would preserve order but discard the absolute degree of overlap between partially matched property labels. This design is therefore closer to similarity matching in cross-modal representation learning (Radford et al., 2021) than to distribution matching.

G Contact Phase Annotation

Each episode is accompanied by post-hoc episode-level annotations of the first and last frames in which the gripper is in physical contact with the object. From these contact-timing annotations we derive four phases for every frame:

Approach. Frames before the first contact frame. The tactile arrays are inactive or carry only ambient noise.

Contact. Frames from first contact through the frame at which the gripper begins to open. This interval covers grasp acquisition and the steady-state hold.

Lift. Frames from gripper-open onset through the last contact frame, during which the object is being released or transferred.

Retreat. Frames after the last contact frame.

For per-phase evaluation and the contact-phase MSE reported in Table 3, only contact frames are used. During training, L_{mono} groups samples by the contact phase.

The same contact-timing annotations are used to construct the contact prior m_t described in Section 3.3. These annotations serve as a phase-timing scaffold for semantic modulation; property identity and action direction remain determined by the language-conditioned representation and policy. The annotation-derived prior $\bar{m}_t$ is set to 0 during approach, 1 during contact, and decays linearly from the last contact frame through the end of lift to 0 at retreat. The online tactile activity score o_t is computed from the maximum taxel activation and the mean contact indicator over the 16-frame tactile window. When the per-episode contact-timing annotations satisfy conservative checks on contact ratio, active taxel ratio, and contact-onset timing, the annotation-validity flag d_t is set to 1 and the blended prior m_t mixes $\bar{m}_t$ and o_t with an annotation blend weight of 0.85, capping the pre-contact online contribution at 0.05. When $d_t = 0$, we fall back to $m_t = o_t$.

In real-robot deployment, post-hoc contact-timing annotations are unavailable, so we use the online tactile activity score alone, i.e., $m_t = o_t$.

H Supplementary Results

All action MSE values are normalized by the training-set action variance $\sigma_{\text{VTLA}}^2 = 808.1$. For pi0.5, its internally computed MSE is rescaled from its own training-space variance to the VTLA-normalized space for approximate aggregate comparison. Because pi0.5 uses a different action representation and normalization interface, its APC-gradient, Swap L2, and negative- Δ metrics should be treated as approximate.

H.1 pi0.5 Evaluation Protocol

We fine-tune the pi0.5 checkpoint (Physical Intelligence et al., 2025) on our dataset using the official OpenPI training pipeline (Physical Intelligence, 2024). Because pi0.5 uses a Chunking-CausalFlowMatching action head with a different action chunking and normalization scheme, a direct latent-space comparison is not available. We therefore adopt the following protocol for each metric:

MSE. We compute MSE in pi0.5’s own normalized action space using the model’s internal de-normalization, then rescale the per-dimension errors to the VTLA-normalized space (scaling factor $\sigma_{\text{VTLA}}^2 = 808.1$) for aggregate comparison. The rescaling is approximate because pi0.5’s action chunking predicts a 50-timestep trajectory from which we extract the immediate action, whereas VTLA baselines predict a single 6-DoF target.

APC gradient and Swap L2. We evaluate pi0.5 on the same held-out slices and APC-level prompts as the VTLA models. The APC gradient $\Delta(\text{L2}, \text{L0})$ and Swap L2 are computed from the rescaled MSE values following the same definitions in Section 3.1. Because the rescaling is approximate, these metrics are reported with a dagger in Table 1 and should be interpreted as indicative rather than strictly comparable.

Negative- Δ count. We count the number of held-out task slices for which the rescaled APC gradient is negative, following the same criterion as for the VTLA models.

Real-robot evaluation. pi0.5 is deployed on the same SO-101 platform with the same camera and tactile sensors. At inference time, pi0.5 receives only the RGB image, the robot state, and the instruction; it does not use tactile input. Each task is evaluated with 20 trials under the same success criteria as the VTLA baselines.

This protocol gives pi0.5 a fair operational comparison while acknowledging that its action representation, tokenizer, and training pipeline differ systematically from the VTLA family. The offline metrics for pi0.5 are best read as a sanity check: a strong external VLA should not be worse than a vision-only baseline on these tasks, and the real-robot results confirm that pi0.5 performs competitively on visually clear tasks while struggling on tactile-critical ones.

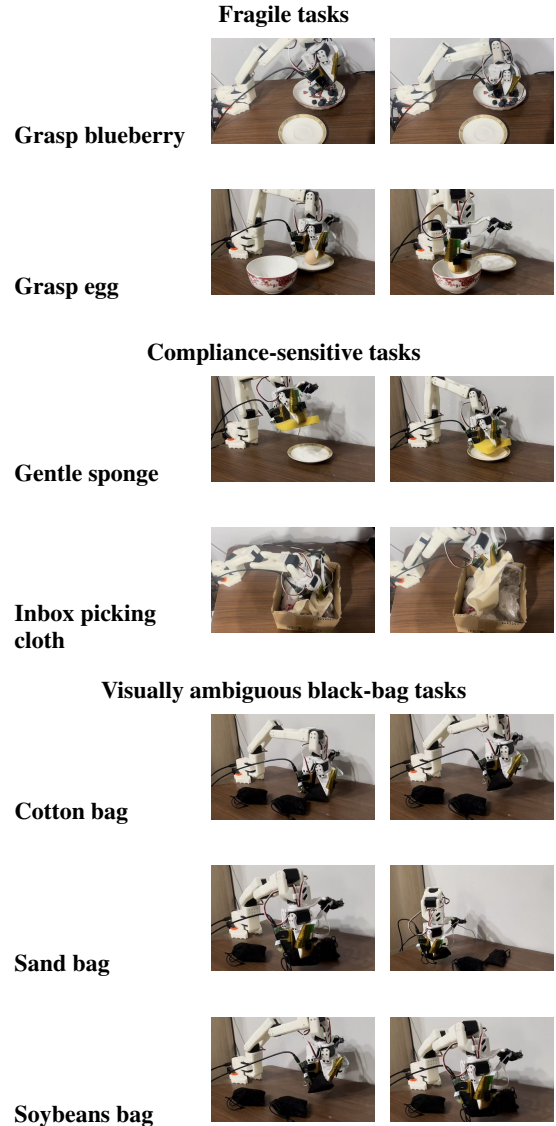


Figure 4: Robot tasks. Each row shows one task name on the left and two representative center-cropped rollout frames on the right. The tasks are grouped into three categories: fragile tasks, where the robot must handle breakable objects gently; compliance-sensitive tasks, where the robot must adapt contact to deformable objects; and visually ambiguous black-bag tasks, where post-contact tactile evidence is needed to distinguish the material and select the appropriate behavior.