GUIDEDVTLA: WHERE PROACTIVE TACTILE GUIDANCE MEETS REACTIVE TACTILE REFINEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

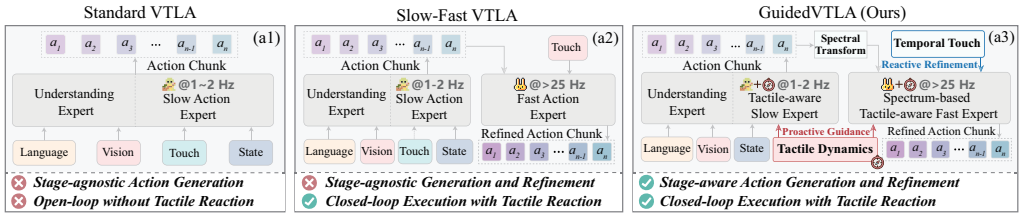
Multi-stage contact-rich tasks are challenging since robots must coordinate global stage-wise execution with local real-time contact reaction, with each global contact stage dictating when and how to react locally. However, existing methods typically use touch as reactive feedback for action generation or refinement in a stage-agnostic manner, limiting their ability to adapt generation and refinement strategies across distinct stages. In this work, we argue that touch is like a coin with two complementary sides: locally, it serves as *reactive* feedback conveying contact information, while globally, its temporal dynamics act as a *proactive* indicator offering contact stage awareness. To unlock the latter, we explore the temporal dynamics prior of optical tactile sensing, and discover that jointly leveraging the cumulative tactile motions and their transient differences can provide robust stage guidance for both action generation and refinement. Building on this insight, we propose GuidedVTLA, a slow-fast VTLA model that integrates proactive tactile guidance with reactive tactile refinement. Specifically, we utilize the shared proactive tactile guidance to coordinate both slow and fast action experts. The slow expert uses this guidance to dynamically route a stage-aware Mixture-of-Experts, adapting action generation strategies across contact stages. The fast expert uses temporal tactile cues to refine the action chunk during execution, with the shared guidance modulating refinement strength. Furthermore, we find that action trajectory energy across stages is mainly concentrated in a low-frequency subspace. This observation motivates us to perform tactile-driven action refinement within this subspace using a compact set of spectral residual coefficients, promoting temporally smooth refinement within each action chunk. Extensive experiments involving multi-stage contact-rich tasks on two real-world robot platforms and in simulation demonstrate the superior performance and robustness of GuidedVTLA.

1 INTRODUCTION

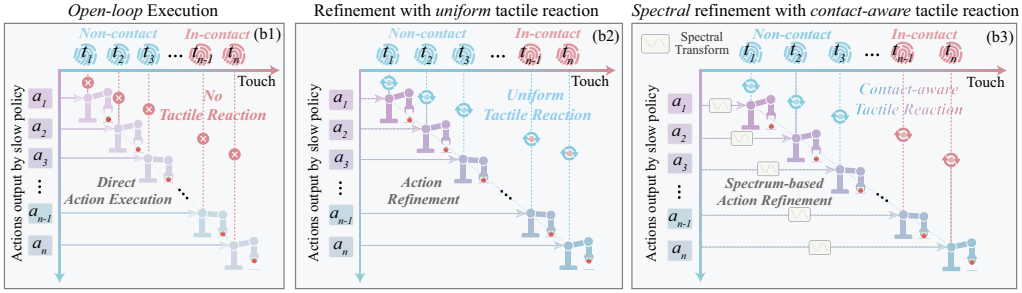
Executing complex multi-stage contact-rich tasks, such as assembling multiple gears, requires robots to coordinate stage-wise execution with real-time contact responses as they progress through multiple stages and contact state transitions. However, most existing methods focus on either multi-stage contact-sparse tasks or short-horizon contact-rich tasks, while practical yet challenging multi-stage contact-rich manipulation remains relatively underexplored. The core challenge lies in the tight coupling between two temporal scales: ❶ Globally, robots need to adapt their action execution strategies across successive stages and contact transitions over longer horizons. ❷ Locally, they need to react to rapid contact changes by refining ongoing actions in real time. Importantly, these two requirements are coupled as the global stage determines when and how the robot should react locally.

To tackle contact-rich manipulation, integrating vision for global environment perception with touch for local contact feedback has become the dominant paradigm. Existing visuo-tactile methods can be broadly divided into two categories: Visuo-Tactile-Action (VTA) models and Visuo-Tactile-Language-Action (VTLA) models. VTA models (Wu et al., 2025; Huang et al., 2024a) evolve from conventional visual policies, typically fusing visual and tactile features into an action expert for action chunk prediction. However, due to the lack of high-level semantic priors and limited model capacity, these methods struggle with multi-stage tasks. To remedy this, VTLA models (Zhang et al., 2026a; Huang et al., 2025b) incorporate touch into VLA-style architectures and achieve significant progress through their powerful semantic understanding. Yet most methods treat touch as a reactive signal throughout the entire manipulation process and employ a monolithic execution strategy for all contact stages, failing to adapt their execution strategies to different stages. Moreover, once action chunks are output, they are directly executed open-loop, preventing timely reactions to contact changes [See Fig. 1(b1)].

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.



(a) Illustration of three existing VTLA paradigms



(b) Comparisons of three paradigms for action chunk refinement

Figure 1: (a) Three VTLA paradigms and (b) their refinement strategies. Standard VTLAs treat touch purely reactively, relying on a stage-agnostic slow expert with open-loop execution that prevents real-time tactile feedback. Slow-fast VTLAs introduce a fast expert to close this loop, yet both experts remain stage-agnostic, applying a uniform tactile reaction regardless of the contact stage. In contrast, GuidedVTLA couples proactive tactile guidance with reactive tactile refinement. By using the tactile dynamics as shared guidance for both experts, we can perform stage-aware action generation and refinement. We perform refinement in the low-frequency domain to promote temporal smoothness.

To close this feedback loop, several methods (Xue et al., 2025; Chen et al., 2025c) adopt a slow-fast paradigm: a slow expert outputs action chunks, while a fast expert refines them during execution using real-time tactile feedback [See Fig. 1(a2)]. Although these methods improve responsiveness, both action generation and refinement lack stage awareness: the slow expert generates action chunks without explicit stage guidance, while the fast expert refines them without explicitly accounting for the current contact stage [See Fig. 1(b2)]. Overall, these methods primarily exploit touch for reactive feedback, leaving its potential for stage guidance underexplored. This raises a question: *How can we infer the contact stage online without annotation, and use it to guide both the slow and fast experts?*

To address the above issues, we propose **GuidedVTLA**, a slow-fast Visuo-Tactile-Language-Action model that integrates *proactive* tactile guidance with *reactive* tactile refinement for multi-stage contact-rich manipulation [See Fig. 1(a3)]. Our key insight is that tactile cues are like a coin with two complementary sides. At the micro level, they serve as *reactive* feedback that conveys local contact information during interactions, which existing methods have already extensively exploited. At the macro level, their temporal dynamics can act as a *proactive* stage indicator, implicitly encoding global contact stages and their transitions without any annotation, yet this aspect remains underexplored.

In this work, we are the first to integrate this complementary dual nature into a unified framework. Specifically, we explore the temporal dynamics of optical tactile sensing, which captures contact-induced gel deformation utilizing an internal camera. We discover that *jointly leveraging cumulative tactile motion and its transient differences can provide a robust proactive indicator for global contact stage awareness*. Concretely, cumulative tactile motion captures overall deformation relative to the initial no-contact state, providing longer-term contact context, while its transient differences between consecutive motions offer short-term contact changes. Together, these complementary signals provide shared guidance for the slow and fast experts to enable stage-aware action generation and refinement.

Building on this shared guidance, we coordinate the slow and fast experts in a stage-aware manner. For the slow expert, we introduce a stage-aware Mixture-of-Experts layer routed by the guidance to adapt the action generation strategies across stages. For the fast expert, we use temporal tactile cues to refine actions during execution under the guidance, which modulates refinement strength. Furthermore, we find that action trajectory energy is mainly concentrated in a low-frequency subspace, motivating us to perform action refinement in the low-frequency spectrum, promoting temporally smooth refinement within each action chunk [See Fig. 1(b3)]. Under this unified framework, the two roles of touch complement each other: proactive guidance offers global stage awareness while reactive feedback supplies local fine-grained refinement, jointly empowering GuidedVTLA to achieve robust multi-stage contact-rich manipulation. Overall, our contributions are summarized as follows:

- We reveal that touch has a complementary dual role, serving as both *reactive* feedback signals providing local contact information and *proactive* stage indicators offering global stage awareness.
- We discover that jointly leveraging cumulative tactile motion and its frame-to-frame difference can serve as a robust stage indicator to provide global stage awareness without any external annotation.
- We propose GuidedVTLA, a slow-fast VTLA model that integrates proactive tactile guidance with reactive tactile refinement. The slow expert uses a stage-aware MoE routed by this guidance to adapt action generation across stages, while the fast expert refines action chunks using temporal tactile feedback under the shared guidance. Motivated by the concentration of action energy in low-frequency modes, we perform refinement in this subspace to promote temporal smoothness.
- We compare GuidedVTLA with existing baselines on both real-world and simulated contact-rich tasks. Extensive experiments demonstrate the superior performance and robustness of our method.

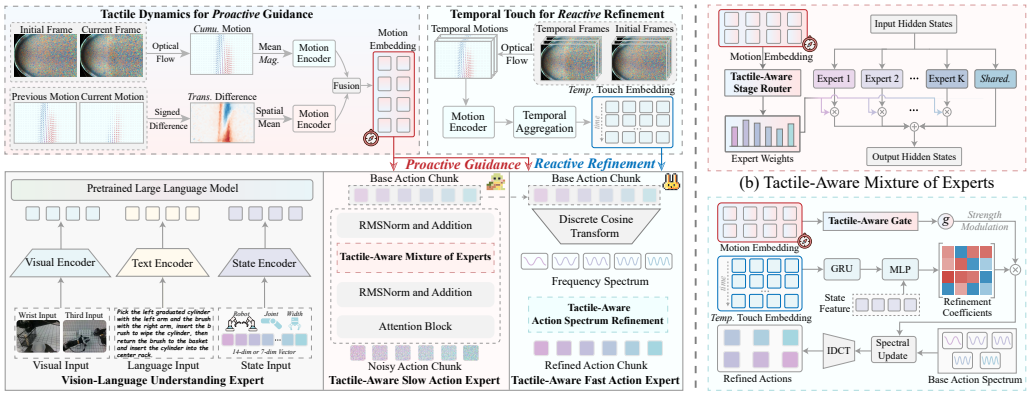
2 Related Work

Visual-Tactile Robotic Manipulation is essential for enabling robots to perform real-world contact-rich tasks (Feng et al., 2025; 2026; Luo et al., 2025; Liu et al., 2025b; Li et al., 2024; Xu et al., 2025). Vision typically provides global perception, while touch offers local contact feedback during interactions. To capture tactile signals, a variety of sensing technologies have been developed (Lin et al., 2023; Zhang et al., 2022; Bhirangi et al., 2021; Lin et al., 2022; Taylor et al., 2022). Among these, optical sensors (Yuan et al., 2017; Xen; Dai) are widely used due to their high spatial resolution and multi-dimensional force sensing capability, which serve as the utilized tactile sensor in this work.

Architecturally, existing visual-tactile methods can be categorized into visual-tactile-action (VTA) and visual-tactile-language-action (VTLA) models. VTA models (Chen et al., 2025b;c; Zhu et al., 2025; Feng et al., 2024; Huang et al., 2024a) typically extend purely vision-based policies (Chi et al., 2025; Zhao et al., 2024) by fusing visual and tactile features prior to action chunk prediction. For instance, Gu et al. (2025) and Wu et al. (2025) concatenate multi-modal features as input to a transformer backbone, whereas Xu et al. (2026) propose a Mixture-of-Transformer architecture for multi-modal feature fusion. However, these methods are inherently limited by model capacity and lack of high-level semantic understanding, making it difficult to handle complex multi-stage tasks. With the rise of Vision-Language-Action models (Intelligence et al., 2025; Kim et al., 2025; 2024; Zhang et al., 2026d), which inherit the strong semantic understanding capabilities, recent works incorporate touch into VLA architectures, yielding VTLA models (Zhang et al., 2026c;g; Morissette et al., 2026; Yuan et al., 2026). Most methods typically treat tactile cues as image-like features that are simply concatenated with visual and linguistic features before being fed into a VLM (Huang et al., 2025b; Zhang et al., 2025). Another line injects touch into the action expert to steer action generation (Zhang et al., 2026g; Huang et al., 2026; Li et al., 2026; Team et al., 2026). However, most methods treat touch as a purely *reactive* signal, applying a stage-agnostic strategy across the whole task and overlooking its *proactive* role. In this work, we propose a slow-fast VTLA model that integrates proactive tactile guidance with reactive tactile refinement for multi-stage contact-rich manipulation.

Slow-Fast Visuo-Tactile Policies (Wang et al., 2026; Zhou et al., 2026) have emerged as a promising paradigm for contact-rich manipulation. Xue et al. (2025) first introduce this slow-fast paradigm, decoding a coarse action latent from the slow action expert into refined actions at a high rate from streaming touch. Niu et al. (2026) split the denoising process of a flow-matching policy into two stages, where a fast tactile-driven expert performs the final denoising steps using real-time touch. Zhang et al. (2026f) refine low-rate action chunks with high-rate observed and predicted touch. Yet, they still remain stage-agnostic. In contrast, we coordinate both action experts with a shared proactive tactile guidance, enabling stage-aware action generation and refinement. Besides, we find the low-frequency priors of action trajectories and perform action refinement in the frequency domain.

Frequency-Based Action Modeling (Zhang et al., 2026b) has emerged as an effective strategy to capture the temporal structure of actions. For instance, Pertsch et al. (2025) tokenize and compress action sequences in the frequency domain. Cao et al. (2026) produce actions in a coarse-to-fine manner across frequency components. Wang (2026) guides action diffusion along sub-frequency manifolds. Tan et al. (2024) encode time-varying control features in the Fourier frequency domain. Yet, these methods are open-loop and vision-only. In contrast, we are the first to perform tactile-driven action refinement in the frequency domain, where confining refinement to the low-frequency coefficients keeps the refined action temporally smooth and requires a few coefficients to be predicted.



(a) Slow-Fast VTLA with Proactive Guidance and Reactive Refinement

(b) Tactile-Aware Mixture of Experts

(c) Tactile-Aware Action Spectral Refinement

Figure 2: Overview of GuidedVTLA. (a) Slow-Fast VTLA with proactive guidance and reactive refinement. Tactile dynamics serve as shared guidance for both action experts, enabling stage-aware action generation and refinement. The fast expert utilizes temporal touch for reactive refinement to better capture contact trends. (b) The slow expert adopts a Tactile-aware MoE guided by tactile dynamics to adapt action generation across stages. (c) The fast expert performs action refinement in the low-frequency spectrum, promoting temporally smooth refinement within each action chunk.

3 Slow-Fast VTLA: Proactive Guidance and Reactive Refinement

We propose GuidedVTLA, a slow-fast Visuo-Tactile-Language-Action model that integrates proactive tactile guidance with reactive tactile refinement, as shown in Fig. 2. Our key insight is that tactile cues play a complementary dual role: beyond the *reactive* feedback that conveys local contact information, their temporal dynamics further act as a *proactive* indicator that implicitly encodes global stage and transition cues. Grounded in this insight, we discover that jointly leveraging the cumulative tactile motion and its transient differences can provide a robust contact stage indicator without any annotation, motivating us to propose *shared tactile guidance* that can steer both the slow and fast action experts to perform stage-aware action generation and refinement, respectively (Section 3.1).

Specifically, for the slow expert, we propose a tactile-aware MoE layer routed by the shared guidance, enabling proactive stage-aware action generation (Section 3.2). For the fast expert, we use temporal tactile cues to refine actions during execution under the shared guidance. Motivated by the discovery that action trajectory energy is mainly concentrated in low-frequency modes across stages, we perform action refinement within a low-frequency subspace to promote temporal smoothness (Section 3.3).

3.1 Shared Proactive Stage Awareness from Tactile Dynamics

Existing approaches to stage awareness often rely on manual annotations or labels generated by external models, which can be costly to obtain or unreliable under visual ambiguity. Meanwhile, existing visuo-tactile methods primarily use touch for reactive contact feedback, leaving its potential for proactive stage guidance underexplored. To exploit this potential, we investigate tactile dynamics and derive a contact-stage indicator that provides shared guidance for both the slow and fast experts.

Representing Tactile Dynamics via Cumulative Motion and Transient Difference. To explore how tactile dynamics evolve across different stages and optical tactile sensors, we conduct an analytical experiment using two different tactile sensors mounted on a gripper, as shown in Fig. 3. Figure 3(a) shows the visual images and tactile images during a multi-stage contact-rich manipulation task (*Gears Assembly*). Although the two sensors produce visually different tactile images, both capture contact-induced gel deformation as the interaction progresses. To characterize this deformation and its temporal evolution, we compute cumulative tactile motion and the transient difference of its magnitude for each sensor. Given the initial frame $\mathbf{F}_0$, the previous frame $\mathbf{F}_{t-1}$, and the current frame $\mathbf{F}_t$, we first estimate the cumulative motion of $\mathbf{F}_t$ and $\mathbf{F}_{t-1}$ relative to $\mathbf{F}_0$ via optical flow:

$$\mathbf{M}_t^{\text{cumu}} = \text{Flow}(\mathbf{F}_0, \mathbf{F}_t), \qquad \mathbf{M}_{t-1}^{\text{cumu}} = \text{Flow}(\mathbf{F}_0, \mathbf{F}_{t-1}), \tag{1}$$

Here, $\text{Flow}(\cdot)$ is an efficient optical flow estimation algorithm (Kroeger et al., 2016). $\mathbf{M}_t^{\text{cumu}}$ and $\mathbf{M}_{t-1}^{\text{cumu}}$ represent the cumulative tactile motions at time t and $t-1$, respectively. To quantitatively measure the cumulative motions, we compute the spatial average of their motion magnitudes:

$$\bar{M}_t^{\text{cumu}} = \frac{1}{HW} \sum_{i,j} \left\| \mathbf{M}_t^{\text{cumu}}(i,j) \right\|_2, \qquad \bar{M}_{t-1}^{\text{cumu}} = \frac{1}{HW} \sum_{i,j} \left\| \mathbf{M}_{t-1}^{\text{cumu}}(i,j) \right\|_2, \tag{2}$$

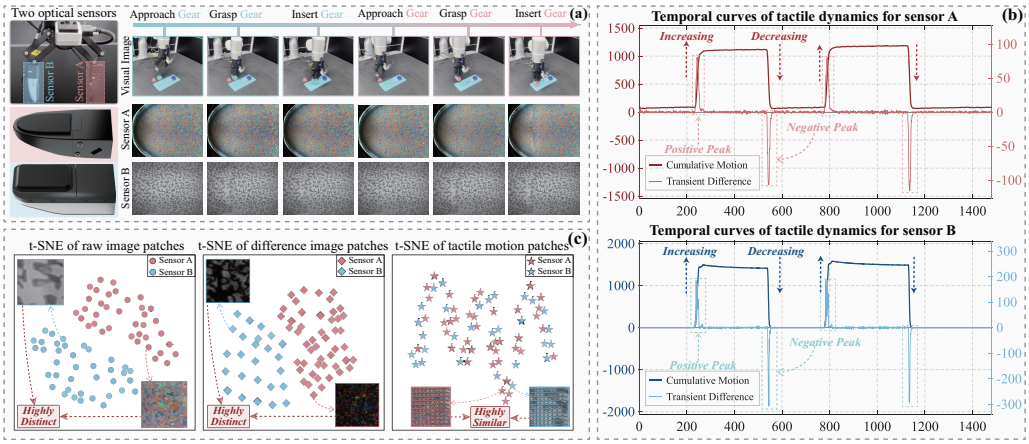


Figure 3: Analysis of tactile dynamics using optical sensors. (a) Visual images and tactile images from two various sensors across stages. (b) shows temporal tactile dynamics curves for both sensors, indicating that tactile dynamics enable stage awareness. (c) shows t-SNE maps of various representations for both sensors, highlighting the potential of tactile dynamics for cross-sensor universality.

where H and W denote the height and width of the motion fields. To capture short-term changes in deformation, we compute the signed difference between consecutive mean motion magnitudes:

$$\bar{M}_t^{\text{tran}} = \bar{M}_t^{\text{cumu}} - \bar{M}_{t-1}^{\text{cumu}}. \tag{3}$$

We intentionally retain the *signed* difference to distinguish the transition direction: positive for contact establishment and negative for release, whereas absolute values would conflate these opposing events.

How Do Tactile Dynamics Achieve Proactive Stage Awareness? To relate these two measures to global contact stages, we compute their temporal curves for both sensors in Fig. 3(b). The cumulative motion magnitude and its transient difference exhibit complementary patterns. During approach, both remain near zero. Upon initial contact, the cumulative magnitude rises with a positive peak in the transient difference. During stable contact, the cumulative magnitude remains elevated, while the difference returns to near zero. Upon release, the cumulative magnitude decreases with a negative transient peak. Neither measure alone fully characterizes these contact stages. Similar cumulative magnitudes can occur during increasing or decreasing deformation, while a near-zero transient difference can indicate either stable contact or no contact. Together, they provide complementary cues for stage awareness by combining longer-term contact context with short-term contact changes.

Sensor-Agnostic Tactile Dynamics. Beyond their relevance to contact stages, we further study whether tactile dynamics are consistent across sensors. Figure 3(b) illustrates that the temporal curves of both sensors exhibit similar shapes despite differences in tactile appearance. We further compare tactile dynamics with two image-based representations (raw image and tactile difference image) by visualizing their low-dimensional distributions via t-SNE in Fig. 3(c). The image-based representations form largely separate clusters for the two sensors, while tactile dynamics show greater overlap. This consistency stems from the shared physical basis of gel deformation: tactile dynamics capture deformation relative to a reference state, reducing sensitivity to sensor-specific appearance. These observations support the potential of tactile dynamics for cross-sensor stage guidance.

Tactile Dynamics as Proactive Stage Guidance. The two measures provide stage-discriminative cues and exhibit consistent patterns across sensors. We encode them using lightweight tactile motion encoders, then concatenate and fuse the two embeddings into a compact guidance representation:

$$\mathbf{T}_t^{\text{pro}} = \mathcal{F}\big(\big[E_{\text{tran}}(\bar{M}_t^{\text{tran}}) \,;\, E_{\text{cumu}}(\bar{M}_t^{\text{cumu}}) \big] \big) \in \mathbb{R}^d, \tag{4}$$

where $\mathcal{F}$ is the fusion layer and $[\,\cdot\,;\,\cdot\,]$ denotes concatenation. The resulting embedding $\mathbf{T}_t^{\text{pro}}$ encodes global contact stage cues and serves as shared guidance for the following slow and fast action experts.

3.2 Slow Action Expert with Tactile-aware Action Generation

Multi-stage contact-rich tasks typically require distinct execution strategies at various stages. However, most methods adopt a monolithic action module to generate actions, which struggles to adapt action generation in a stage-aware manner. To address this issue, we propose a tactile-aware slow action expert powered by proactive stage awareness. Specifically, we incorporate tactile-aware MoE modules whose routing is conditioned on the shared tactile guidance, enabling stage-aware action generation.

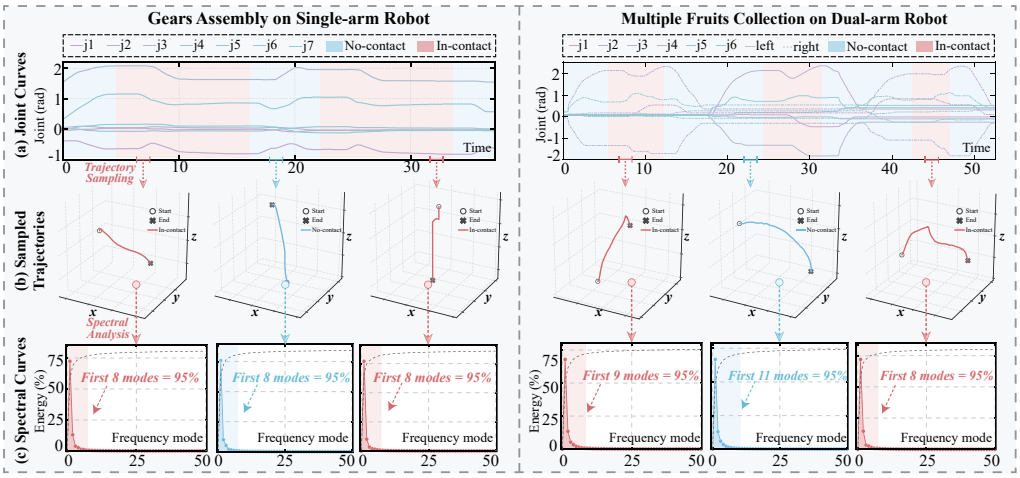


Figure 4: Spectral analysis of action trajectories across contact stages. (a) shows temporal curves of joint positions for two robotic arms in two tasks. (b) presents 3D visualizations of randomly sampled action trajectories across stages. (c) performs spectral analysis of action trajectories across stages, revealing that they are concentrated in a low-frequency subspace.

Multimodal Observations and Action Generation. At time step t, the robot receives an instruction L and an observation tuple $\mathcal{O}_t = (\mathbf{I}_t, \mathbf{F}_0, \mathbf{F}_t, \mathbf{F}_{t-1}, \mathbf{p}_t)$, where $\mathbf{I}_t$ contains multi-view RGB images, $\mathbf{F}_0$, $\mathbf{F}_t$, and $\mathbf{F}_{t-1}$ represent the initial contact-free, current, and previous tactile frames, and $\mathbf{p}_t$ is the proprioceptive state. A pretrained VLM backbone encodes vision, language, and proprioception into an action query $\mathbf{x}_t^A$, while the tactile frames provide the shared guidance $\mathbf{T}_t^{\text{pro}}$ proposed in Section 3.1. Conditioned on these inputs, the slow action expert generates a base action chunk $\mathbf{A}_t^{\text{base}} \in \mathbb{R}^{H_a \times d_a}$ at a low update rate, where H_a denotes the prediction horizon and d_a denotes the action dimension.

Tactile-aware Mixture-of-Experts. To inject the proactive guidance into action generation, we introduce a tactile-aware MoE module in each transformer block of the slow action expert, as shown in Fig. 2(b). Each module comprises a shared expert $\mathcal{E}_{\text{sh}}$ for general action generation and K stage-conditioned experts $\{\mathcal{E}_k\}_{k=1}^K$ specialized for distinct stages. A router R maps the guidance $\mathbf{T}_t^{\text{pro}}$ to routing weights that are broadcast across action tokens and reused in all tactile-aware MoE layers:

$$\mathbf{w}_t = [\, w_t^1, \ldots, w_t^K \,] = \text{softmax}(R(\mathbf{T}_t^{\text{pro}}))\,, \tag{5}$$

where w_t^k denotes the gating weight for the k-th stage expert. The action query is subsequently transformed by a combination of the shared expert and the weighted stage experts:

$$\tilde{\mathbf{x}}_t^A = \mathcal{E}_{\text{sh}}(\mathbf{x}_t^A) + \sum_{k=1}^{K} w_t^k \mathcal{E}_k(\mathbf{x}_t^A), \tag{6}$$

The resulting representations are used to generate the base action chunk $\mathbf{A}_t^{\text{base}}$ via flow matching. This design provides a common tactile-aware routing signal across all action tokens within each chunk, while allowing multiple experts to contribute through soft weighting rather than discrete stage assignments. The slow expert incorporates stage cues into action generation. However, contact states may change during execution, motivating the fast action expert introduced next for online refinement.

3.3 Fast Action Expert with Tactile-aware Action Spectrum Refinement

Most existing methods execute generated action chunks open-loop, limiting their ability to react to intra-chunk contact events. To close the loop, existing slow-fast methods use real-time tactile feedback to refine action chunks during execution, but lack explicit contact-stage guidance. To address this limitation, we introduce a tactile-aware fast action expert that uses temporal tactile feedback for refinement, with shared proactive guidance modulating refinement strength. Furthermore, we observe that action trajectory energy is predominantly concentrated in low-frequency modes, which motivates us to perform action refinement in a low-frequency subspace rather than directly in the time domain.

Spectral Property of Action Trajectories. To explore the spectral properties of action trajectories across stages, we conduct an analytical experiment on two robotic platforms during multi-stage contact-rich tasks, as shown in Fig. 4. Figure 4(a) shows the temporal curves of the joint positions of two robotic arms, partitioned into contact and non-contact stages. We sample action chunks from

Table 1: Quantitative comparison of success rates against baselines on real multi-stage contact-rich tasks. Policies are evaluated over 20 trials per task, with the best performance highlighted in bold.

Multi-Stage Contact-Rich Tasks on Dual-Arm ALOHA Robot									
Methods	**Cylinder Clean & Collection**			**Fragile Fruits Collection**			**Fragile Chips Collection**		
	Grasp Objects →	Insert & Clean →	Insert Back	1st Fruit Grape →	2nd Fruit Tomato →	3rd Fruit Blueberry	Intact Grasp →	First Chip →	Second Chip
ACT [RSS2023]	85.0	15.0	*5.0*	20.0	15.0	*0.0*	25.0	20.0	*10.0*
$\pi_{0.5}$ [CoRL2025]	**100.0**	40.0	*35.0*	70.0	60.0	*40.0*	75.0	70.0	*70.0*
RDP [RSS2025]	90.0	25.0	*20.0*	30.0	30.0	*20.0*	50.0	45.0	*40.0*
ViTacMotor [ECCV2026]	90.0	30.0	*30.0*	35.0	25.0	*10.0*	30.0	30.0	*20.0*
TacVLA [IROS2026]	95.0	45.0	*45.0*	70.0	70.0	*45.0*	80.0	75.0	*75.0*
ForceVLA [NIPS2025]	**100.0**	35.0	*30.0*	75.0	60.0	*50.0*	65.0	65.0	*65.0*
GuidedVTLA [This paper]	**100.0**	**65.0**	***55.0***	**80.0**	**75.0**	***60.0***	**90.0**	**85.0**	***85.0***
Multi-Stage Contact-Rich Tasks on Single-Arm Flexiv Robot									
Methods	**Pick and Insert Plug**			**Gears Assembly**			**Pick and Insert Bulb**		
	Grasp Plug →	Align Plug →	Insert Plug	Grasp Gears →	First Gear →	Second Gear	Grasp Bulb →	Align Bulb →	Insert Bulb
ACT [RSS2023]	20.0	5.0	*0.0*	10.0	0.0	*0.0*	10.0	0.0	*0.0*
$\pi_{0.5}$ [CoRL2025]	**100.0**	65.0	*60.0*	**100.0**	60.0	*60.0*	**100.0**	55.0	*45.0*
RDP [RSS2025]	60.0	40.0	*20.0*	50.0	15.0	*10.0*	55.0	25.0	*10.0*
ViTacMotor [ECCV2026]	30.0	10.0	*5.0*	20.0	5.0	*0.0*	35.0	15.0	*15.0*
TacVLA [IROS2026]	**100.0**	75.0	*70.0*	**100.0**	70.0	*65.0*	**100.0**	**65.0**	***60.0***
ForceVLA [NIPS2025]	90.0	65.0	*55.0*	85.0	55.0	*50.0*	90.0	45.0	*45.0*
GuidedVTLA [This paper]	**100.0**	**80.0**	***80.0***	95.0	**80.0**	***75.0***	**100.0**	60.0	***60.0***

each stage, with the corresponding 3D trajectories shown in Fig. 4(b), and analyze their spectral properties in Fig. 4(c). Across contact and non-contact stages, the spectrum mainly concentrates in the *low-frequency subspace*. This pattern is consistent with the smooth temporal evolution of the observed action trajectories, motivating us to refine action chunks within a low-frequency subspace.

Temporal Tactile Encoding for Reactive Feedback. To enable refinement in the low-frequency subspace, we encode temporal tactile cues to capture contact evolution beyond a single frame. We use the cumulative motion fields, which preserve spatial deformation magnitudes and directions beyond the scalar summaries used for proactive guidance. Given the motion fields from the recent N steps, a spatial encoder E_{s} first extracts features from each field. A temporal aggregator TAgg then combines these features into a window-level embedding $\mathbf{z}_t$. To incorporate contact history beyond this window, a GRU updates the reactive tactile embedding $\mathbf{T}_t^{\mathrm{rea}}$ using $\mathbf{z}_t$ and the previous hidden state:

$$\begin{aligned} \mathbf{z}_t &= \mathrm{TAgg}\big(\{\, E_{\mathrm{s}}(\mathbf{M}_i^{\mathrm{cumu}}) \,\}_{i=t-N+1}^{t}\big), \\ \mathbf{T}_t^{\mathrm{rea}} &= \mathrm{GRU}\big(\mathbf{T}_{t-1}^{\mathrm{rea}},\ \mathbf{z}_t\big) \in \mathbb{R}^{d_r}. \end{aligned} \tag{7}$$

Here, $\mathbf{z}_t$ summarizes tactile dynamics within the recent window, while the recurrent state carries information from earlier control steps. This causal encoding uses only observations available up to time t and provides the tactile feedback for predicting the spectral residuals described below.

Tactile-Aware Action Spectral Refinement. The fast expert uses reactive tactile feedback to refine low-frequency action components, with shared guidance modulating refinement strength, as shown in Fig. 2(c). We first apply a DCT along the temporal axis of each action dimension:

$$\mathbf{C}_t^{\mathrm{base}} = \mathrm{DCT}\big(\mathrm{sg}(\mathbf{A}_t^{\mathrm{base}})\big) \in \mathbb{R}^{H_a \times d_a}, \tag{8}$$

where $\mathrm{sg}(\cdot)$ blocks gradients through the base action chunk, and rows index frequency modes from low to high. An MLP G predicts additive refinement coefficients for the lowest $M < H_a$ modes:

$$\Delta\mathbf{c}_t = G\big([\mathbf{T}_t^{\mathrm{rea}}; \mathbf{p}_t]\big) \in \mathbb{R}^{M \times d_a}, \tag{9}$$

The shared proactive tactile guidance controls the refinement strength through a scalar contact gate:

$$g_t^R = \sigma(\mathbf{W}_g \mathbf{T}_t^{\mathrm{pro}} + b_g) \in (0, 1), \tag{10}$$

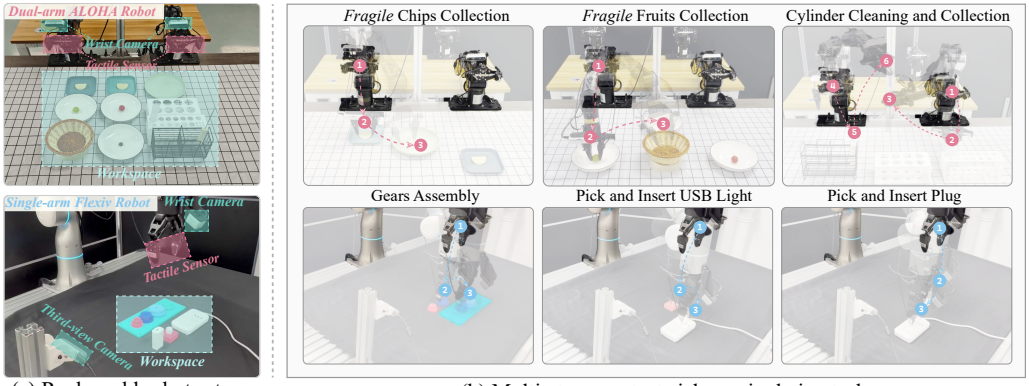
where σ is sigmoid function and $\mathbf{W}_g, b_g$ are parameters. We add the gated refinement coefficients to the lowest M modes of the base spectrum, preserve the remaining modes, and reconstruct the action:

$$\hat{\mathbf{C}}_t[k] = \begin{cases} \mathbf{C}_t^{\mathrm{base}}[k] + g_t^R \Delta\mathbf{c}_t[k], & 0 \le k < M, \\ \mathbf{C}_t^{\mathrm{base}}[k], & M \le k < H_a, \end{cases} \qquad \hat{\mathbf{A}}_t = \mathrm{IDCT}(\hat{\mathbf{C}}_t). \tag{11}$$

Here, $[k]$ indexes frequency modes, with $k = 0$ denoting the constant mode, and $\hat{\mathbf{C}}_t, \hat{\mathbf{A}}_t \in \mathbb{R}^{H_a \times d_a}$ are the refined spectrum and action chunk, respectively. Using only M low-frequency modes promotes temporally smooth refinement within each chunk while requiring just $M \times d_a$ predicted coefficients.

4 Experimental Results

4.1 Experimental Setup



(a) Real world robot setup (b) Multi-stage contact-rich manipulation tasks

Figure 5: Illustration of experimental setup. (a) Two robotic platforms equipped with optical tactile sensors and multi-view cameras. (b) Multi-stage contact-rich tasks performed on the two platforms.

Training Details. GuidedVTLA is built on PaliGemma with a Gemma-based action expert. The slow and fast experts are trained jointly with $\mathcal{L} = \mathcal{L}_{\text{flow}} + \mathcal{L}_{\text{fast}}$, where $\mathcal{L}_{\text{flow}}$ optimizes flow matching on the action chunk and $\mathcal{L}_{\text{fast}} = \|\hat{\mathbf{A}}_t - \mathbf{A}_t\|_2^2$ regresses the refined chunk. We train each task for 20k steps on 4 NVIDIA A800 GPUs, with batch size of 64, learning rate 5×10^{-5}. The details of the configuration are provided in the appendix A.1.

Table 2: Quantitative comparison on UniVTAC. Results with * are from the FTP-1 paper. The **best** and **second-best** results are highlighted.

Methods	Lift Bottle	Pull-out Key	Lift Can	Put Bottle	Insert Hole	Insert Tube	Insert HDMI	Grasp Classify	**Avg. [1–6]**	**Avg. All**
ACT [RSS2023]	25.0	32.0	43.0	20.0	10.0	**83.0**	15.0	44.0	35.5	34.0
$\pi_{0.5}$ *(Base)* [CoRL2025]	**100.0**	35.0	6.0	34.0	25.0	74.0	8.0	49.0	45.7	41.4
UniVTAC-ACT [CoRL2026]	65.0	41.0	20.0	30.0	40.0	68.0	**28.0**	**100.0**	44.0	**49.0**
Tactile-VLA* [arXiv2026]	**97.0**	32.0	15.0	10.0	41.0	56.0	–	–	41.8	–
FTP-1* [CoRL2026]	**97.0**	**48.0**	**65.0**	**47.0**	**64.0**	79.0	–	–	**66.7**	–
GuidedVTLA [This paper]	75.0	**99.0**	**76.0**	**66.0**	**96.0**	**97.0**	**21.0**	**74.0**	**84.8**	**75.5**
vs. prev. SoTA	−25.0	+51.0	+11.0	+19.0	+32.0	+14.0	−7.0	−26.0	+18.2	+26.5
vs. Base Model	−25.0	+64.0	+70.0	+32.0	+71.0	+23.0	+13.0	+25.0	+39.2	+34.1

Real-World and Simulation Experimental Setup. For real-world evaluation, we evaluate on two robotic platforms: a dual-arm ALOHA robot and a single-arm Flexiv robot, both equipped with optical tactile sensors and multi-view cameras [See Fig. 5(a)]. We design six multi-stage contact-rich tasks across the two embodiments, including *Cylinder Cleaning and Collection*, *Fragile Fruits Collection*, *Fragile Chips Collection*, *Plug Insertion*, *Gear Assembly* and *USB Bulb Insertion* [See Fig. 5(b)]. The details of tasks are provided in the appendix A.2. For simulation, we evaluate on a simulated visuo-tactile manipulation benchmark UniVTAC (Chen et al., 2026) across eight contact-rich tasks (*e.g.* insert tube and insert hole). The details of each task are provided in the appendix A.3.

Baselines. We compare GuidedVTLA with (1) Tactile-free models: ACT (Zhao et al., 2023) and $\pi_{0.5}$ (Intelligence et al., 2025), (2) VTA models: RDP (Xue et al., 2025), ViTacMotor (Xu et al., 2026), (3) VTLA models: TacVLA (Zhang et al., 2026c), ForceVLA (Yu et al., 2025). For TacVLA, as the code is not publicly available, we reproduce it based on $\pi_{0.5}$. For ForceVLA, we replace the force input with tactile input to form a VTLA model. For simulation, we additionally compare against Tactile-VLA (Huang et al., 2025b), FTP-1 (Yuan et al., 2026) and UniVTAC-ACT (Chen et al., 2026).

4.2 Comparison Results

Comparison on Real-world Tasks. Table 1 compares GuidedVTLA with baselines on real-world tasks. Visual-only policies struggle with fragile object collection and precision assembly due to missing force perception and visual occlusions. Though visuo-tactile methods incorporate touch, their stage-agnostic action execution is affected by tactile noise in non-contact phases, impairing action generation and refinement. While VTLA models achieve robust initial grasping leveraging pre-trained priors, their open-loop chunk execution cannot compensate for intra-chunk slip, accumulating errors during precision insertion. Besides, they lack stage awareness to adaptively adjust action strategies. In contrast, GuidedVTLA achieves superior performance by harnessing tactile dynamics for proactive stage awareness and temporal touch for reactive spectrum-based action refinement.

Comparison on Simulation Tasks. We further benchmark GuidedVTLA against other methods on the UniVTAC benchmark in Tab. 2. GuidedVTLA achieves the highest average success rate and the best performance on five tasks. Notably, on *Insert Hole* and *Insert Tube*, which involve contact-state transitions and require continuous pose adjustment, GuidedVTLA substantially outperforms the baselines by combining stage-aware tactile guidance with high-frequency tactile feedback.

Table 3: Ablation study of the GuidedVTLA components.

Base	TA-Slow	TA-Fast	Gears Assembly	Fruits Collection
✓			60.0	40.0
✓	✓		70.0	45.0
✓		✓	65.0	55.0
✓	✓	✓	**75.0**	**60.0**

Table 4: Impact of proactive tactile guidance on both experts.

$\mathbf{T}^{\text{pro}} \rightarrow$ Slow	$\mathbf{T}^{\text{pro}} \rightarrow$ Fast	Gears Assembly	Fruits Collection
		50.0	45.0
✓		60.0	45.0
	✓	65.0	50.0
✓	✓	**75.0**	**60.0**

Table 5: Effectiveness of low-frequency tactile refinement.

Refinement Space	Gear Assembly	Fruits Collection
w/o Refinement	65.0	50.0
Time Domain	40.0	45.0
Freq., Full Band	60.0	55.0
Freq., Low Band	**75.0**	**60.0**

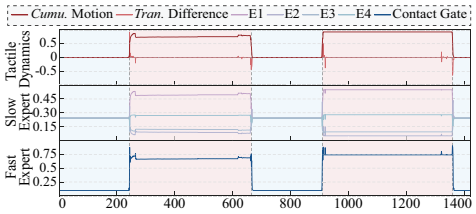


Figure 6: Effectiveness of tactile dynamics in guiding both experts for stage awareness.

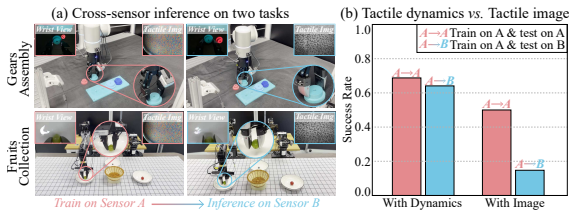


Figure 7: Cross-sensor universality of tactile dynamics *vs.* tactile images on *Gears Asm.* & *Fruits Collection*.

4.3 Ablation Study and Discussion

How Does the Tactile-Aware Slow Expert Facilitate Action Generation? We further validate the role of the tactile-aware slow expert in action generation. Table 3 shows that removing the tactile-aware MoE layer from the slow expert leads to a drop in success rate on both gears assembly and fruits collection. Both tasks involve multiple contact stages and transitions. Without stage guidance, failures concentrate at these transitions, such as when the gear is inserted into the socket, confirming the effectiveness of the tactile-aware MoE in facilitating stage-aware action generation.

How Does the Tactile-Aware Fast Expert Improve Contact-Rich Manipulation? Table 3 shows that removing the fast expert likewise causes a drop in success rate. The primary role of fast expert is to refine the action chunk generated by the slow expert from streaming tactile feedback during execution. Without it, the action chunk is executed in an open-loop manner without online adjustment to dynamic contact changes. Consequently, the robot struggles with delicate contact adaptations, such as adjusting the gripper pose upon gear misalignment or modulating the gripper width based on force when grasping fragile fruits, which demonstrates the necessity of the fast expert for action refinement.

How Do Tactile Dynamics Guide Action Generation and Refinement? Table 4 shows that ablating the guidance from either end, by reverting the slow expert to standard MoE routing or by removing the tactile gating of the fast expert, consistently degrades performance. Furthermore, Figure 6 visualizes the temporal trajectories of the tactile dynamics along with the expert weights of the slow expert and the gating of the fast expert during gear assembly. Both follow the tactile dynamics across stages, confirming its effectiveness in guiding stage-aware action generation and refinement.

Cross-Sensor Universality of the Tactile Dynamics. We further discuss the cross-sensor universality of tactile dynamics. Figure 7 shows that our model trained with tactile sensor A (Neo) can still perform well when deployed on sensor B (Dai), with only a minor drop in success rate. In contrast, replacing the tactile dynamics with raw images for proactive guidance and reactive refinement leads to a significant degradation, confirming the cross-sensor generalization capability of our method.

Why Refine in Frequency Domain Rather than in Time Domain? Table 5 compares different refinement spaces. Time-domain refinement performs worse than no refinement, possibly because unconstrained updates introduce rapid temporal variations and trajectory jitter during contact. Low-frequency refinement achieves the best performance on both tasks, outperforming full-band spectral refinement. These results demonstrate the effectiveness of restricting refinement bandwidth, consistent with its role in discouraging high-frequency fluctuations and promoting temporal smoothness.

5 Conclusion

We propose GuidedVTLA, a slow-fast VTLA that unifies proactive tactile guidance with reactive tactile refinement. Cumulative tactile motion and its transient differences jointly provide stage guidance for both experts: the slow expert adapts execution strategies through a stage-aware MoE, while the fast expert refines actions using temporal tactile feedback. Refinement operates in a compact low-frequency subspace to improve efficiency and promote temporal smoothness. Extensive experiments demonstrate the superior performance and robustness of the proposed method.

AI USE STATEMENT

We only used AI to polish and correct the text and grammar in the main text.

REFERENCES

Daimon optical tactile sensor, dm-tac w2. https://www.dmrobot.com/product/p1/dm-tacw2.html, DM-Tac W2.

Neote ai optical tactile sensor, intac s1. https://www.neoteai.com/, InTac S1.

Xense optical tactile sensor, xensesensor. https://www.xenserobotics.com/product/367/detail/9, XenseSensor.

Raunaq Bhirangi, Tess Hellebrekers, Carmel Majidi, and Abhinav Gupta. Reskin: versatile, replaceable, lasting tactile skins. *arXiv preprint arXiv:2111.00071*, 2021.

Jianxin Bi, Kevin Yuchen Ma, Ce Hao, Mike Zheng Shou, and Harold Soh. Vla-touch: Enhancing vision-language-action models with dual-level tactile feedback. *arXiv preprint arXiv:2507.17294*, 2025.

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

Shuxin Cao, Liquan Wang, Walker Byrnes, Yiye Chen, Yilun Du, and Animesh Garg. Hierarchical policy learning via spectral decomposition. *arXiv preprint arXiv:2606.29570*, 2026.

Baijun Chen, Weijie Wan, Tianxing Chen, Xianda Guo, Congsheng Xu, Yuanyang Qi, Haojie Zhang, Longyan Wu, Tianling Xu, Zixuan Li, et al. Univtac: A unified simulation platform for visuo-tactile manipulation data generation, learning, and benchmarking. *arXiv preprint arXiv:2602.10093*, 2026.

Hao Chen, Jiaming Liu, Chenyang Gu, Zhuoyang Liu, Renrui Zhang, Xiaoqi Li, Xiao He, Yandong Guo, Chi-Wing Fu, Shanghang Zhang, et al. Fast-in-slow: A dual-system foundation model unifying fast manipulation within slow reasoning. *arXiv preprint arXiv:2506.01953*, 2025a.

Haonan Chen, Jiaming Xu, Hongyu Chen, Kaiwen Hong, Binghao Huang, Chaoqi Liu, Jiayuan Mao, Yunzhu Li, Yilun Du, and Katherine Driggs-Campbell. Multi-modal manipulation via multi-modal policy consensus. *arXiv preprint arXiv:2509.23468*, 2025b.

Wendi Chen, Han Xue, Yi Wang, Fangyuan Zhou, Jun Lv, Yang Jin, Shirun Tang, Chuan Wen, and Cewu Lu. Implicitrdp: An end-to-end visual-force diffusion policy with structural slow-fast learning. *arXiv preprint arXiv:2512.10946*, 2025c.

Yizhou Chen, Andrea Sipos, Mark Van der Merwe, and Nima Fazeli. Visuo-tactile transformers for manipulation. *arXiv preprint arXiv:2210.00121*, 2022.

Baiye Cheng, Tianhai Liang, Suning Huang, Maanping Shao, Feihong Zhang, Botian Xu, Zhengrong Xue, and Huazhe Xu. Moe-dp: An moe-enhanced diffusion policy for robust long-horizon robotic manipulation with skill decomposition and failure recovery. *arXiv preprint arXiv:2511.05007*, 2025.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Lutfi Eren Erdogan, Nicholas Lee, Sehoon Kim, Suhong Moon, Hiroki Furuta, Gopala Anumanchipalli, Kurt Keutzer, and Amir Gholami. Plan-and-act: Improving planning of agents for long-horizon tasks. *arXiv preprint arXiv:2503.09572*, 2025.

Huang Fang, Mengxi Zhang, Heng Dong, Wei Li, Zixuan Wang, Qifeng Zhang, Xueyun Tian, Yucheng Hu, and Hang Li. Robix: A unified model for robot interaction, reasoning and planning. *arXiv preprint arXiv:2509.01106*, 2025.

Ruoxuan Feng, Di Hu, Wenke Ma, and Xuelong Li. Play to the score: Stage-guided dynamic multi-sensory fusion for robotic manipulation. *arXiv preprint arXiv:2408.01366*, 2024.

Ruoxuan Feng, Jiangyu Hu, Wenke Xia, Tianci Gao, Ao Shen, Yuhao Sun, Bin Fang, and Di Hu. Anytouch: Learning unified static-dynamic representation across multiple visuo-tactile sensors. *arXiv preprint arXiv:2502.12191*, 2025.

Ruoxuan Feng, Yuxuan Zhou, Siyu Mei, Dongzhan Zhou, Pengwei Wang, Shaowei Cui, Bin Fang, Guocai Yao, and Di Hu. Anytouch 2: General optical tactile representation learning for dynamic tactile perception. *arXiv preprint arXiv:2602.09617*, 2026.

Shijia Ge, Yinxin Zhang, Shuzhao Xie, Weixiang Zhang, Mingcai Zhou, and Zhi Wang. Vggt-dp: Generalizable robot control via vision foundation models. *arXiv preprint arXiv:2509.18778*, 2025.

Abraham George, Selam Gano, Pranav Katragadda, and Amir Barati Farimani. Vital pretraining: Visuo-tactile pretraining for tactile and non-tactile manipulation policies. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 258–264. IEEE, 2025.

Ningquan Gu, Kazuhiro Kosuge, and Mitsuhiro Hayashibe. Tactilealoha: Learning bimanual manipulation with tactile sensing. *IEEE Robotics and Automation Letters*, 2025.

Zihao He, Hongjie Fang, Jingjing Chen, Hao-Shu Fang, and Cewu Lu. Foar: Force-aware reactive policy for contact-rich robotic manipulation. *IEEE Robotics and Automation Letters*, 2025.

Binghao Huang, Yixuan Wang, Xinyi Yang, Yiyue Luo, and Yunzhu Li. 3d-vitac: Learning fine-grained manipulation with visuo-tactile sensing. *arXiv preprint arXiv:2410.24091*, 2024a.

Binghao Huang, Jie Xu, Iretiayo Akinola, Wei Yang, Balakumar Sundaralingam, Rowland O'Flaherty, Dieter Fox, Xiaolong Wang, Arsalan Mousavian, Yu-Wei Chao, et al. Vt-refine: Learning bimanual assembly with visuo-tactile feedback via simulation fine-tuning. *arXiv preprint arXiv:2510.14930*, 2025a.

Haoxu Huang, Fanqi Lin, Yingdong Hu, Shengjie Wang, and Yang Gao. Copa: General robotic manipulation through spatial constraints of parts with foundation models. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 9488–9495. IEEE, 2024b.

Jialei Huang, Shuo Wang, Fanqi Lin, Yihang Hu, Chuan Wen, and Yang Gao. Tactile-vla: unlocking vision-language-action model's physical knowledge for tactile generalization. *arXiv preprint arXiv:2507.09160*, 2025b.

Yuzhe Huang, Pei Lin, Wanlin Li, Daohan Li, Jiajun Li, Jiaming Jiang, Chenxi Xiao, and Ziyuan Jiao. Tactile-force alignment in vision-language-action models for force-aware manipulation. *arXiv preprint arXiv:2601.20321*, 2026.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Xuhui Kang, Tongxuan Tian, Sung-Wook Lee, Binghao Huang, Yunzhu Li, and Yen-Ling Kuo. Learning force-regulated manipulation with a low-cost tactile-force-controlled gripper. *arXiv preprint arXiv:2602.10013*, 2026.

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

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

Till Kroeger, Radu Timofte, Dengxin Dai, and Luc Van Gool. Fast optical flow using dense inverse search. In *European conference on computer vision*, pp. 471–488. Springer, 2016.

Hao Li, Yizhi Zhang, Junzhe Zhu, Shaoxiong Wang, Michelle A Lee, Huazhe Xu, Edward Adelson, Li Fei-Fei, Ruohan Gao, and Jiajun Wu. See, hear, and feel: Smart sensory fusion for robotic manipulation. *arXiv preprint arXiv:2212.03858*, 2022.

Jinzhou Li, Tianhao Wu, Jiyao Zhang, Zeyuan Chen, Haotian Jin, Mingdong Wu, Yujun Shen, Yaodong Yang, and Hao Dong. Adaptive visuo-tactile fusion with predictive force attention for dexterous manipulation. *arXiv preprint arXiv:2505.13982*, 2025a.

Shoujie Li, Zihan Wang, Changsheng Wu, Xiang Li, Shan Luo, Bin Fang, Fuchun Sun, Xiao-Ping Zhang, and Wenbo Ding. When vision meets touch: A contemporary review for visuotactile sensors from the signal processing perspective. *IEEE Journal of Selected Topics in Signal Processing*, 18 (3):267–287, 2024.

Xiaoqi Li, Muhe Cai, Jiadong Xu, Juan Zhu, Hongwei Fan, Yan Shen, Guangrui Ren, and Hao Dong. At-vla: Adaptive tactile injection for enhanced feedback reaction in vision-language-action models. *arXiv preprint arXiv:2605.07308*, 2026.

Yuyang Li, Yinghan Chen, Zihang Zhao, Puhao Li, Tengyu Liu, Siyuan Huang, and Yixin Zhu. Simultaneous tactile-visual perception for learning multimodal robot manipulation. *arXiv preprint arXiv:2512.09851*, 2025b.

Changyi Lin, Ziqi Lin, Shaoxiong Wang, and Huazhe Xu. Dtact: A vision-based tactile sensor that measures high-resolution 3d geometry directly from darkness. *arXiv preprint arXiv:2209.13916*, 2022.

Changyi Lin, Han Zhang, Jikai Xu, Lei Wu, and Huazhe Xu. 9dtact: A compact vision-based tactile sensor for accurate 3d shape reconstruction and generalizable 6d force estimation. *IEEE Robotics and Automation Letters*, 9(2):923–930, 2023.

Fanqi Lin, Ruiqian Nai, Yingdong Hu, Jiacheng You, Junming Zhao, and Yang Gao. Onetwovla: A unified vision-language-action model with adaptive reasoning. *arXiv preprint arXiv:2505.11917*, 2025.

Fangchen Liu, Chuanyu Li, Yihua Qin, Jing Xu, Pieter Abbeel, and Rui Chen. Vitamin: Learning contact-rich tasks through robot-free visuo-tactile manipulation interface. *arXiv preprint arXiv:2504.06156*, 2025a.

Jason Jingzhou Liu, Yulong Li, Kenneth Shaw, Tony Tao, Ruslan Salakhutdinov, and Deepak Pathak. Factr: Force-attending curriculum training for contact-rich policy learning. *arXiv preprint arXiv:2502.17432*, 2025b.

Jiaming Liu, Hao Chen, Pengju An, Zhuoyang Liu, Renrui Zhang, Chenyang Gu, Xiaoqi Li, Ziyu Guo, Sixiang Chen, Mengzhen Liu, et al. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. *arXiv preprint arXiv:2503.10631*, 2025c.

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. *arXiv preprint arXiv:2410.07864*, 2024.

Zhuoyang Liu, Jiaming Liu, Jiadong Xu, Nuowei Han, Chenyang Gu, Hao Chen, Kaichen Zhou, Renrui Zhang, Kai Chin Hsieh, Kun Wu, et al. Mla: A multisensory language-action model for multimodal understanding and forecasting in robotic manipulation. *arXiv preprint arXiv:2509.26642*, 2025d.

Shan Luo, Nathan F Lepora, Wenzhen Yuan, Kaspar Althoefer, Gordon Cheng, and Ravinder Dahiya. Tactile robotics: An outlook. *IEEE Transactions on Robotics*, 2025.

Charlotte Morissette, Amin Abyaneh, Wei-Di Chang, Anas Houssaini, David Meger, Hsiu-Chin Lin, Jonathan Tremblay, and Gregory Dudek. Tactile modality fusion for vision-language-action models. *arXiv preprint arXiv:2603.14604*, 2026.

Zehao Ni, Yonghao He, Lingfeng Qian, Jilei Mao, Fa Fu, Wei Sui, Hu Su, Junran Peng, Zhipeng Wang, and Bin He. Vo-dp: Semantic-geometric adaptive diffusion policy for vision-only robotic manipulation. *arXiv preprint arXiv:2510.15530*, 2025.

Dantong Niu, Zhuoyang Liu, Zekai Wang, Boning Shao, Zhao-Heng Yin, Anirudh Pai, Yuvan Sharma, Stefano Saravalle, Ruijie Zheng, Jing Wang, et al. T-rex: Tactile-reactive dexterous manipulation. *arXiv preprint arXiv:2606.17055*, 2026.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Lucy Xiaoyang Shi, Brian Ichter, Michael Equi, Liyiming Ke, Karl Pertsch, Quan Vuong, James Tanner, Anna Walling, Haohuan Wang, Niccolo Fusai, et al. Hi robot: Open-ended instruction following with hierarchical vision-language-action models. *arXiv preprint arXiv:2502.19417*, 2025.

Hengkai Tan, Songming Liu, Kai Ma, Chengyang Ying, Xingxing Zhang, Hang Su, and Jun Zhu. Fourier controller networks for real-time decision-making in embodied learning. *arXiv preprint arXiv:2405.19885*, 2024.

Yikai Tang, Haoran Geng, Sheng Zang, Pieter Abbeel, and Jitendra Malik. Visual-geometry diffusion policy: Robust generalization via complementarity-aware multimodal fusion. *arXiv preprint arXiv:2511.22445*, 2025.

Ian H Taylor, Siyuan Dong, and Alberto Rodriguez. Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 10781–10787. IEEE, 2022.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

NeoteAI Team, Fudan TEAI Team, et al. n_0-vtla: Scaling vision-tactile-language-action model with latent tactile tokens. *arXiv preprint arXiv:2607.23782*, 2026.

Mark Van der Merwe, Mohamad Louai Shehab, Jayjun Lee, Youngsun Wi, Yinpei Dai, Dmitry Berenson, and Nima Fazeli. Tap-vla: Tactile annotation prompting for vision language action models. *arXiv preprint arXiv:2606.29089*, 2026.

Chenxi Wang, Hongjie Fang, Hao-Shu Fang, and Cewu Lu. Rise: 3d perception makes real-world robot imitation simple and effective. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2870–2877. IEEE, 2024.

Junlin Wang. Frequency-guided action diffusion via sub-frequency manifold traversal. *arXiv preprint arXiv:2605.27919*, 2026.

Yi Wang, Wendi Chen, Zimo Wen, Han Xue, Xueqi Li, Wenye Yu, Zhijie Chen, Hao Yang, Jun Lv, Chuan Wen, et al. Never too late for force: Accelerating vla post-training with reactive force injection. *arXiv preprint arXiv:2607.14236*, 2026.

Longyan Wu, Checheng Yu, Jieji Ren, Li Chen, Yufei Jiang, Ran Huang, Guoying Gu, and Hongyang Li. Freetacman: Robot-free visuo-tactile data collection system for contact-rich manipulation. *arXiv preprint arXiv:2506.01941*, 2025.

Shengqi Xu, Guojin Zhong, Yang Liu, Fanjie Wang, Hu Luo, Hanyu Zhou, Weiyao Zhang, Ziyi Ye, Zuxuan Wu, and Yu-Gang Jiang. Seeing touch from motion: A unified modality-aware visuo-tactile policy with tactile motion correlation. *arXiv preprint arXiv:2606.29941*, 2026.

Yue Xu, Litao Wei, Pengyu An, Qingyu Zhang, and Yong-Lu Li. exumi: Extensible robot teaching system with action-aware task-agnostic tactile representation. *arXiv preprint arXiv:2509.14688*, 2025.

Han Xue, Jieji Ren, Wendi Chen, Gu Zhang, Yuan Fang, Guoying Gu, Huazhe Xu, and Cewu Lu. Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation. *arXiv preprint arXiv:2503.02881*, 2025.

Zhutian Yang, Caelan Garrett, Dieter Fox, Tomás Lozano-Pérez, and Leslie Pack Kaelbling. Guiding long-horizon task and motion planning with vision language models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 16847–16853. IEEE, 2025.

Guo Ye, Zexi Zhang, Xu Zhao, Shang Wu, Haoran Lu, Shihan Lu, and Han Liu. Learning to feel the future: Dreamtacvla for contact-rich manipulation. *arXiv preprint arXiv:2512.23864*, 2025.

Jiawen Yu, Hairuo Liu, Qiaojun Yu, Jieji Ren, Ce Hao, Haitong Ding, Guangyu Huang, Guofan Huang, Yan Song, Panpan Cai, et al. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation. *arXiv preprint arXiv:2505.22159*, 2025.

Chengbo Yuan, Zicheng Zhang, Mingjie Zhou, Wendi Chen, Yi Wang, Zhuoyang Liu, Dantong Niu, Shuo Wang, Hui Zhang, Wenkang Zhang, et al. Ftp-1: A generalist foundation tactile policy across tactile sensors for contact-rich manipulation. *arXiv preprint arXiv:2606.13102*, 2026.

Wenzhen Yuan, Siyuan Dong, and Edward H Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762, 2017.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. *arXiv preprint arXiv:2403.03954*, 2024.

Yanjie Ze, Zixuan Chen, Wenhao Wang, Tianyi Chen, Xialin He, Ying Yuan, Xue Bin Peng, and Jiajun Wu. Generalizable humanoid manipulation with 3d diffusion policies. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2873–2880. IEEE, 2025.

Chaofan Zhang, Peng Hao, Xiaoge Cao, Xiaoshuai Hao, Shaowei Cui, and Shuo Wang. Vtla: Vision-tactile-language-action model with preference learning for insertion manipulation. *arXiv preprint arXiv:2505.09577*, 2025.

Chaofan Zhang, Peng Hao, Xiaoge Cao, Xiaoshuai Hao, Shaowei Cui, and Shuo Wang. Vtla: Vision-tactile-language-action model with preference learning for insertion manipulation. *Biomimetic Intelligence and Robotics*, pp. 100333, 2026a.

Jinhui Zhang, Haimin Yao, Jiaying Mo, Songyue Chen, Yu Xie, Shenglin Ma, Rui Chen, Tao Luo, Weisong Ling, Lifeng Qin, et al. Finger-inspired rigid-soft hybrid tactile sensor with superior sensitivity at high frequency. *Nature communications*, 13(1):5076, 2022.

Jiyao Zhang, Zimu Han, Junhan Wang, Xionghao Wu, Shihong Lin, Jinzhou Li, Hongwei Fan, Ruihai Wu, Dongjiang Li, and Hao Dong. Hipolicy: Hierarchical multi-frequency action chunking for policy learning. *arXiv preprint arXiv:2604.06067*, 2026b.

Kaidi Zhang, Heng Zhang, Zhengtong Xu, Zhiyuan Zhang, Md Rakibul Islam Prince, Xiang Li, Xiaojing Han, Yuhao Zhou, Arash Ajoudani, and Yu She. Tacvla: Contact-aware tactile fusion for robust vision-language-action manipulation. *arXiv preprint arXiv:2603.12665*, 2026c.

Likui Zhang, Tao Tang, Zhihao Zhan, Xiuwei Chen, Zisheng Chen, Jianhua Han, Jiangtong Zhu, Pei Xu, Hang Xu, Hefeng Wu, et al. Atomicvla: Unlocking the potential of atomic skill learning in robots. *arXiv preprint arXiv:2603.07648*, 2026d.

Pengwei Zhang, Bin Xie, Ce Hao, Xinpan Meng, Xinyu Guo, Fang Deng, Long Cheng, and Tiancai Wang. Feeling the unexpected: Restacvla for contact-rich manipulation via residual tactile representation. *arXiv preprint arXiv:2607.03387*, 2026e.

Xidong Zhang, Yichi Zhang, Jiaxin Shi, Fucai Zhu, Siyu Zhu, Michael Yu Wang, Xiaojun Wu, and Weihao Yuan. Unitacvla: Unified tactile understanding and prediction in vision language action models. *arXiv preprint arXiv:2606.31723*, 2026f.

Zhemeng Zhang, Jiahua Ma, Xincheng Yang, Xin Wen, Yuzhi Zhang, Boyan Li, Yiran Qin, Jin Liu, Can Zhao, Li Kang, et al. Touchguide: Inference-time steering of visuomotor policies via touch guidance. *arXiv preprint arXiv:2601.20239*, 2026g.

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

Tony Z Zhao, Jonathan Tompson, Danny Driess, Pete Florence, Kamyar Ghasemipour, Chelsea Finn, and Ayzaan Wahid. Aloha unleashed: A simple recipe for robot dexterity. *arXiv preprint arXiv:2410.13126*, 2024.

Jianyi Zhou, Feiyang Hong, Yunhao Li, Yicheng Zhao, Yongjue Cen, Zirui Liu, Jiakang Huang, Zirui Chen, Ruiyang Zhang, Weizhuo Zhu, et al. Touchworld: A predictive and reactive tactile foundation model for dexterous manipulation. *arXiv preprint arXiv:2607.07287*, 2026.

Zhehua Zhou, Jiayang Song, Kunpeng Yao, Zhan Shu, and Lei Ma. Isr-llm: Iterative self-refined large language model for long-horizon sequential task planning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2081–2088. IEEE, 2024.

Xinyue Zhu, Binghao Huang, and Yunzhu Li. Touch in the wild: Learning fine-grained manipulation with a portable visuo-tactile gripper. *arXiv preprint arXiv:2507.15062*, 2025.

APPENDIX

For a better understanding of the main paper, we provide additional details in this supplementary material, which is organized as follows:

(**Section A**) We provide more details about the experimental settings.

- We provide more implementation and training details of our method in Sec. A.1.
- We provide more details about the real-world experimental setup in Sec. A.2.
- We provide more details about the simulation experimental setup in Sec. A.3.
- We provide more details about the baseline implementations in Sec. A.4.

(**Section B**) We provide additional method and implementation details.

- We describe the optical flow computation for tactile motion estimation in Sec. B.1.
- We provide additional trajectory spectral analyses across tasks in Sec. B.2.
- We detail the shared tactile guidance and slow expert implementation in Sec. B.3.
- We provide implementation details of the fast expert in Sec. B.4.
- We rethink the role of touch through policy design comparisons in Sec. B.5.

(**Section C**) We conduct further discussion and analysis on the performance of our method.

- We discuss the robustness of our method to environmental and object variations in Sec. C.1.
- We study the generalization of our method to object position variations in Sec. C.2.
- We conduct an analytical study on choice of tactile representation for proactive stage awareness in Sec. C.3.
- We further discuss the cross-sensor generalization of the proposed method in Sec. C.4.
- We discuss the limitations and future work of our method in Sec. C.5.

(**Demo**) We provide a video demo to show the superiority and robustness of our method towards multi-stage contact-rich manipulation tasks.

A EXPERIMENTAL SETUP

A.1 IMPLEMENTATION AND TRAINING DETAILS

Model configuration. GuidedVTLA uses PaliGemma with a Gemma-based action expert and is initialized from the public $\pi_{0.5}$ base checkpoint. The slow expert generates a base action chunk with tactile-guided expert routing, while the fast expert refines it using recent tactile observations, with shared guidance controlling the refinement strength through the contact gate. The fast refinement head is initialized to zero, so refinement initially preserves the base prediction up to numerical precision. Table S1 summarizes the key settings.

Joint training. We jointly train both experts with $\mathcal{L} = \mathcal{L}_{\text{flow}} + \mathcal{L}_{\text{fast}}$, where $\mathcal{L}_{\text{flow}}$ is the conditional flow-matching loss and $\mathcal{L}_{\text{fast}} = \|\hat{\mathbf{A}}_t - \mathbf{A}_t\|_2^2$ is the squared error between the refined action chunk

Table S1: Key model and training settings for GuidedVTLA.

Parameter	Setting
Base checkpoint	$\pi_{0.5}$
Action horizon H_a	50
Routing experts in the slow expert K	4
Tactile window length N	5 frames
Retained DCT modes M	8
Shared guidance dimension	64
Fast GRU hidden dimension	128
Fast refinement-head initialization	Zero
Optimizer	AdamW
AdamW (β_1, β_2)	$(0.9, 0.95)$
Peak learning rate	5×10^{-5}
Warm-up steps	1,000
Learning rate after warm-up	Constant
Gradient-norm clip	1.0
Batch size	64
Training steps per task	20,000
Euler steps for training-time base rollout	4
Loss weights $(\mathcal{L}_{\text{flow}}, \mathcal{L}_{\text{fast}})$	$(1, 1)$

$\hat{\mathbf{A}}_t$ and the demonstration chunk $\mathbf{A}_t$. During training, a four-step Euler rollout produces the base chunk, which is detached before refinement. Fast-loss gradients are therefore blocked through the base action generation path, but can still update the shared tactile-guidance network through the refinement pathway. We train each task for 20,000 steps on four NVIDIA A800 GPUs with a batch size of 64 and a peak learning rate of 5×10^{-5}. We use AdamW with linear learning-rate warm-up followed by a constant learning rate.

A.2 Real-world Experimental Setup

For the real-world experimental setup, we evaluate on two robotic platforms: a dual-arm *ALOHA* robot with 6 DoF per arm, and a single-arm *Flexiv* robot with 7 DoF. Each platform is equipped with optical tactile sensors and multi-view visual cameras. We design six multi-stage contact-rich tasks across the two platforms, three on the dual-arm ALOHA and three on the single-arm Flexiv. Each task is decomposed into multiple sub-stages, and we provide the details and success criteria for each task below:

(1) Dual-arm Multi-stage Contact-rich Tasks

Cylinder Cleaning and Collection (350 episodes for training). The left arm picks up a graduated cylinder from the left tube rack and moves it to a central position, while the right arm picks a brush from a basket and inserts it into the cylinder to clean it. The brush is then returned to the basket, and the left arm places the cylinder into a central tube rack. The representative execution results are shown in Fig. S1(a). This task demands precise grasp poses to achieve accurate alignment for both the brush insertion and the placement into the designated rack slot, as well as fine force control during insertion. *Evaluation Criterion*: The task is considered successful if the brush is used to clean the cylinder, the brush is returned to the basket, and the cylinder is placed into the target rack while remaining intact throughout the process.

Multiple Fragile Fruits Collection (350 episodes for training). The two arms sequentially pick three fragile fruits (grape, tomato, blueberry) and place them into one plate without crushing them. The task requires a gentle grasp and a careful release. The representative execution results are shown in Fig. S1(b). Note that the fruits are placed in different positions on the plate to vary initial conditions. *Evaluation Criterion*: The task is considered successful if all three fruits are collected into the plate undamaged.

Multiple Fragile Chips Collection (360 episodes for training). The two arms sequentially pick two fragile potato chips and place them into one plate without breaking them. The task requires a gentle

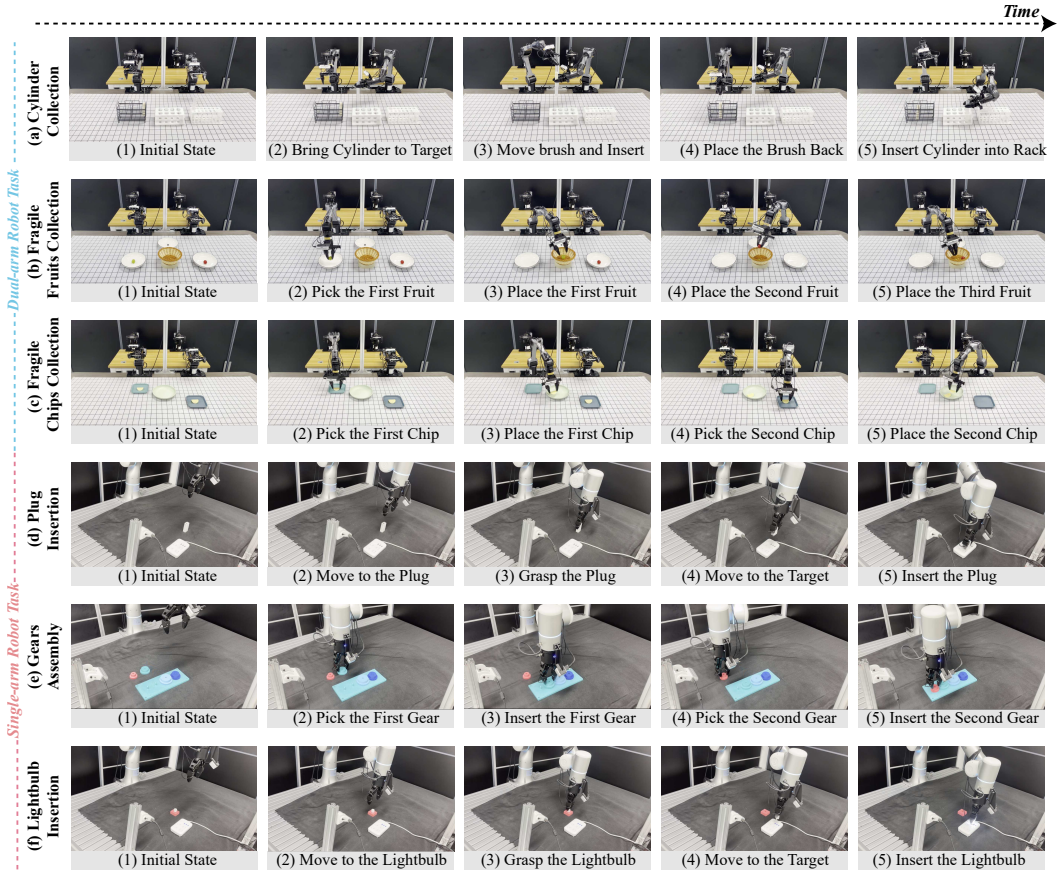


Figure S1: Visualization of representative execution results of real-world multi-stage contact-rich tasks. (a) Cylinder Cleaning and Collection. (b) Multiple Fragile Fruits Collection. (c) Multiple Fragile Chips Collection. (d) Plug Insertion. (e) Gears Assembly. (f) Lightbulb Insertion.

grasp and a careful release. The representative execution results are shown in Fig. S1(c). Note that the chips are placed in different positions on the plate to vary initial conditions. *Evaluation Criterion*: The task is considered successful if both chips are collected into the plate undamaged.

(2) Single-arm Multi-stage Contact-rich Tasks

Plug Insertion (420 episodes for training). The arm grasps a plug and inserts it into a socket, which requires precise alignment and fine adjustment once contact is established. The representative execution results are shown in Fig. S1(d). Note that the plug and socket are placed in different positions to vary initial conditions. *Evaluation Criterion*: The task is considered successful if the plug is fully seated in the socket.

Gears Assembly (430 episodes for training). The arm sequentially grasps two gears, aligns each gear with the shaft, and inserts it once the teeth mesh. The representative execution results are shown in Fig. S1(e). Note that the gears and shaft are placed in different positions to vary initial conditions. *Evaluation Criterion*: The task is considered successful if every gear is correctly aligned and seated on the shaft.

Lightbulb Insertion (450 episodes for training). The arm grasps a USB light bulb and inserts it into a socket, again requiring precise alignment and fine adjustment under contact. The representative execution results are shown in Fig. S1(f). Note that the bulb and socket are placed in different positions to vary initial conditions. *Evaluation Criterion*: The task is considered successful if the lightbulb is fully seated in the socket and the lightbulb is on.

A.3 Simulation Experimental Setup

Task Description. For the simulation experimental setup, we evaluate on the vision-tactile manipulation benchmark UniVTAC (Chen et al., 2026), which contains eight contact-rich tasks and is

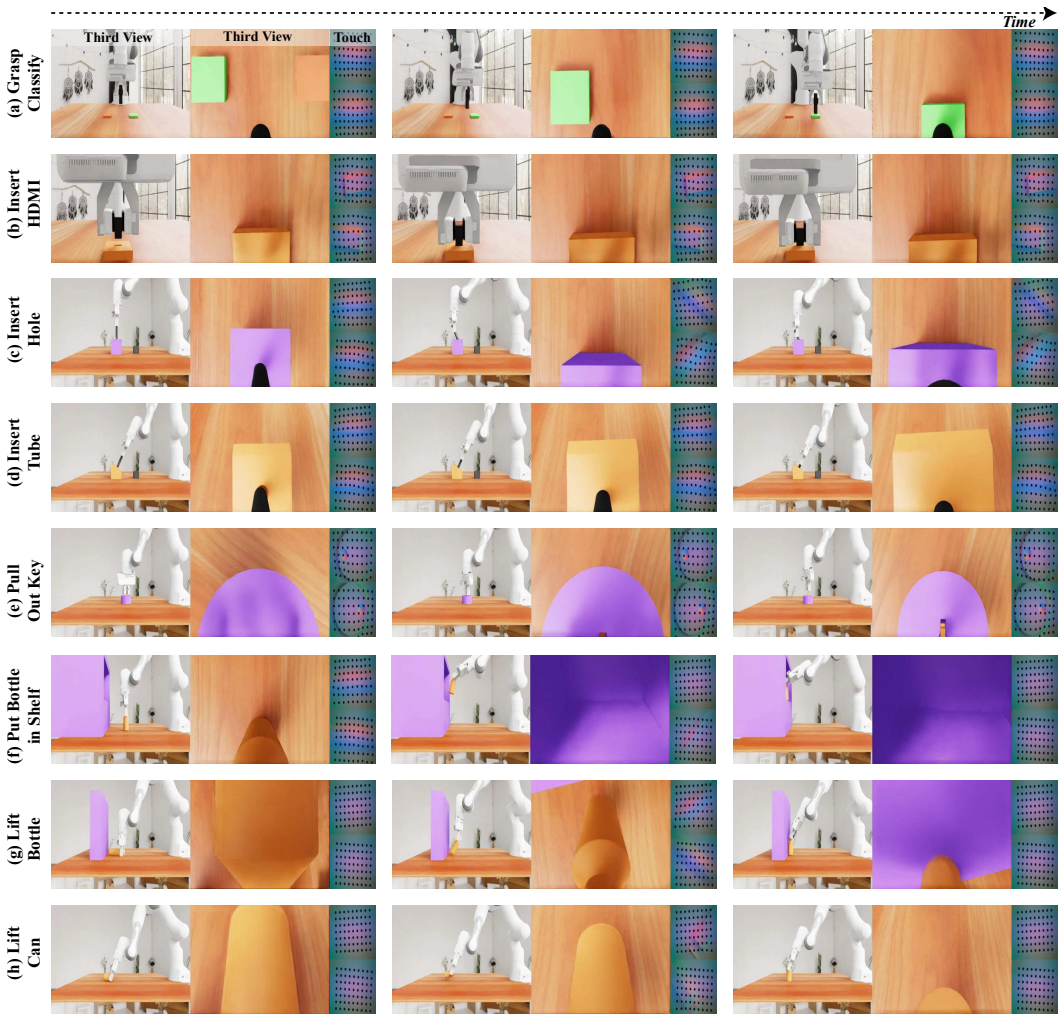


Figure S2: Visualization of representative execution results of simulation contact-rich tasks. (a) Grasp Classify. (b) Insert HDMI. (c) Insert Hole. (d) Insert Tube. (e) Pull Out Key. (f) Put Bottle in Shelf. (g) Lift Bottle. (h) Lift Can.

equipped with a GelSight optical tactile sensor with a uniform sparse marker pattern on the silicone surface. Note that in the simulation, we use the displacement of the markers to represent the tactile dynamics for proactive stage awareness. This design choice is motivated by the fact that, compared to real-world setups that typically employ optical tactile sensors with dense markers, *the sparse marker pattern in simulation makes it challenging to accurately estimate displacements using optical flow methods*. Each task is evaluated 100 times to calculate the success rate. The tasks and their evaluation criteria are detailed as follows:

Grasp Classify (50 episodes for training). Two cylindrical objects share similar visual appearance but differ in surface texture. The robot first performs tactile contact to perceive texture, then classifies each cylinder and places it at the prespecified goal region for its class. The representative execution results are shown in Fig. S2(a). *Evaluation Criterion*: The task is considered successful if both cylinders are correctly classified and placed in their respective goal regions.

Insert HDMI (50 episodes for training). An HDMI connector held by the robot has random rotational offset. The robot aligns and inserts the connector into a fixed slot under rotational uncertainty. The representative execution results are shown in Fig. S2(b). *Evaluation Criterion*: The task is considered successful if the HDMI connector is fully inserted into the slot.

Insert Hole (50 episodes for training). A cube with an inclined hole (60° or 120° orientation) lies on the ground plane. The robot explores the hole's geometry through contact, determines its orientation,

and inserts a test tube into the hole. The representative execution results are shown in Fig. S2(c). *Evaluation Criterion*: The task is considered successful if the test tube is fully inserted into the hole.

Insert Tube (50 episodes for training). A narrow hole (2.05 cm diameter) appears on an inclined surface. The robot inserts a 2.0 cm diameter test tube from a near-center starting position through the tight clearance. The representative execution results are shown in Fig. S2(d). *Evaluation Criterion*: The task is considered successful if the test tube is fully inserted into the hole.

Pull Out Key (50 episodes for training). A key sits in a slot with random initial rotation. The robot rotates the key until sensing mechanical resistance, then pulls it straight out. The representative execution results are shown in Fig. S2(e). *Evaluation Criterion*: The task is considered successful if the key is fully pulled out of the slot.

Put Bottle in Shelf (50 episodes for training). A bottle stands upright before a shelf. The robot grasps the bottle, then positions it into the shelf cavity. The representative execution results are shown in Fig. S2(f). *Evaluation Criterion*: The task is considered successful if the bottle is placed upright in the shelf cavity.

Lift Bottle (50 episodes for training). A bottle rests on the ground plane facing a vertical wall. The robot grasps the bottle and lifts it vertically, keeping its final base within 5 cm of the wall. The representative execution results are shown in Fig. S2(g). *Evaluation Criterion*: The task is considered successful if the bottle is lifted and its base is within 5 cm of the wall.

Lift Can (50 episodes for training). One of three cylindrical cans (4, 5, and 6 cm diameter) lies horizontally on the ground plane. The robot grasps the can and lifts it vertically without slippage. The representative execution results are shown in Fig. S2(h). *Evaluation Criterion*: The task is considered successful if the can is lifted and remains stable without slippage.

A.4 Baseline Details

We compare against three categories of baseline methods on both real-world and simulation setups: (1) tactile-free methods: ACT (Zhao et al., 2023) and $\pi_{0.5}$ (Intelligence et al., 2025); (2) visuo-tactile methods: RDP (Xue et al., 2025) and ViTacMotor (Xu et al., 2026); (3) visuo-tactile-language methods: TacVLA (Zhang et al., 2026c), ForceVLA (Yu et al., 2025), Tactile-VLA (Huang et al., 2025b) and FTP-1 (Yuan et al., 2026). The details of each baseline method are as follows:

(1) Tactile-free Methods.

ACT (Zhao et al., 2023) is a Transformer-based policy that predicts action chunks from RGB observations and proprioception. Without tactile inputs or language conditioning, it serves as a baseline to benchmark the capabilities of purely visual imitation on these tasks. Code is available at https://github.com/tonyzhaozh/act.

$\pi_{0.5}$ (Intelligence et al., 2025) is a pretrained vision-language-action model that unifies high-level semantic prediction and low-level continuous control within a single Mixture-of-Transformers architecture. It first predicts a textual subtask from visual observations and a language instruction and then generates a continuous action chunk using a flow-matching action expert. The original model combines discrete action tokenization during large-scale pretraining with continuous flow matching during post-training. In our comparison, $\pi_{0.5}$ is fine-tuned without tactile observations and serves as the closest tactile-free counterpart to the slow VLA expert used by GuidedVTLA. Code is available at https://github.com/Physical-Intelligence/openpi.

(2) Visuo-Tactile-Action Methods.

RDP (Xue et al., 2025) is a slow-fast visuo-tactile imitation-learning framework rather than a reinforcement-learning method. Its slow branch uses a latent diffusion policy to predict action chunks from visual observations at a relatively low frequency, while its fast branch processes high-frequency tactile or force feedback and refines the latent action representation before decoding it into executable actions. This design enables rapid reaction to contact changes without repeatedly running the full visual diffusion policy. Code is available at https://github.com/xiaoxiaoxh/reactive_diffusion_policy.

Table S2: DIS Optical Flow parameters categorized by algorithm stages

Parameter	Description	Typical Value
	Patch Inverse Search	
θ_{ps}	Patch size	8
θ_{ov}	Patch overlap ratio	0.3
θ_{it}	Inverse search iterations	8
	Multi-scale Pyramid	
θ_{sf}	Finest scale level	2
θ_{ss}	Coarsest scale level	5
θ_{sd}	Downscale factor	2
	Variational Refinement	
δ	Intensity consistency weight	5
γ	Gradient consistency weight	10
α	Smoothness weight	10
θ_{vo}	Outer iterations	$s+1$
θ_{vi}	Inner SOR iterations	5

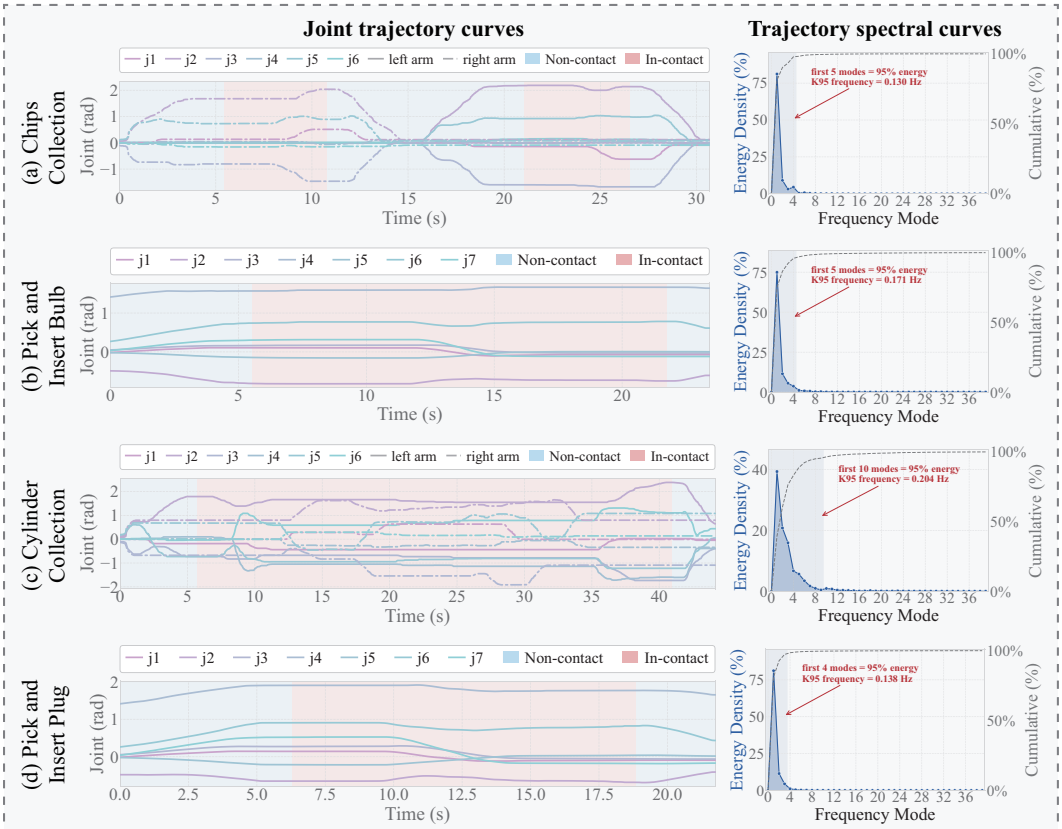


Figure S3: Joint-space trajectories and their mean-centered FFT spectra for four real-world tasks, using one complete trajectory per task. (a) Chips Collection. (b) Pick and Insert Bulb. (c) Cylinder Clean and Collection. (d) Pick and Insert Plug.

ViTacMotor (Xu et al., 2026) introduces a Tactile Motion Correlation (TMC) representation for optical tactile sensing. TMC computes the dot product between transient optical flow, measured between consecutive tactile frames, and cumulative optical flow, measured relative to the initial non-contact frame, thereby distinguishing contact establishment, stable contact, and contact release. ViTacMotor treats visual features, tactile appearance, TMC, and proprioception as separate modalities and fuses them using a Mixture-of-Transformers architecture with shared self-attention but modality-specific projections, feed-forward layers, and normalization. A transformer decoder subsequently

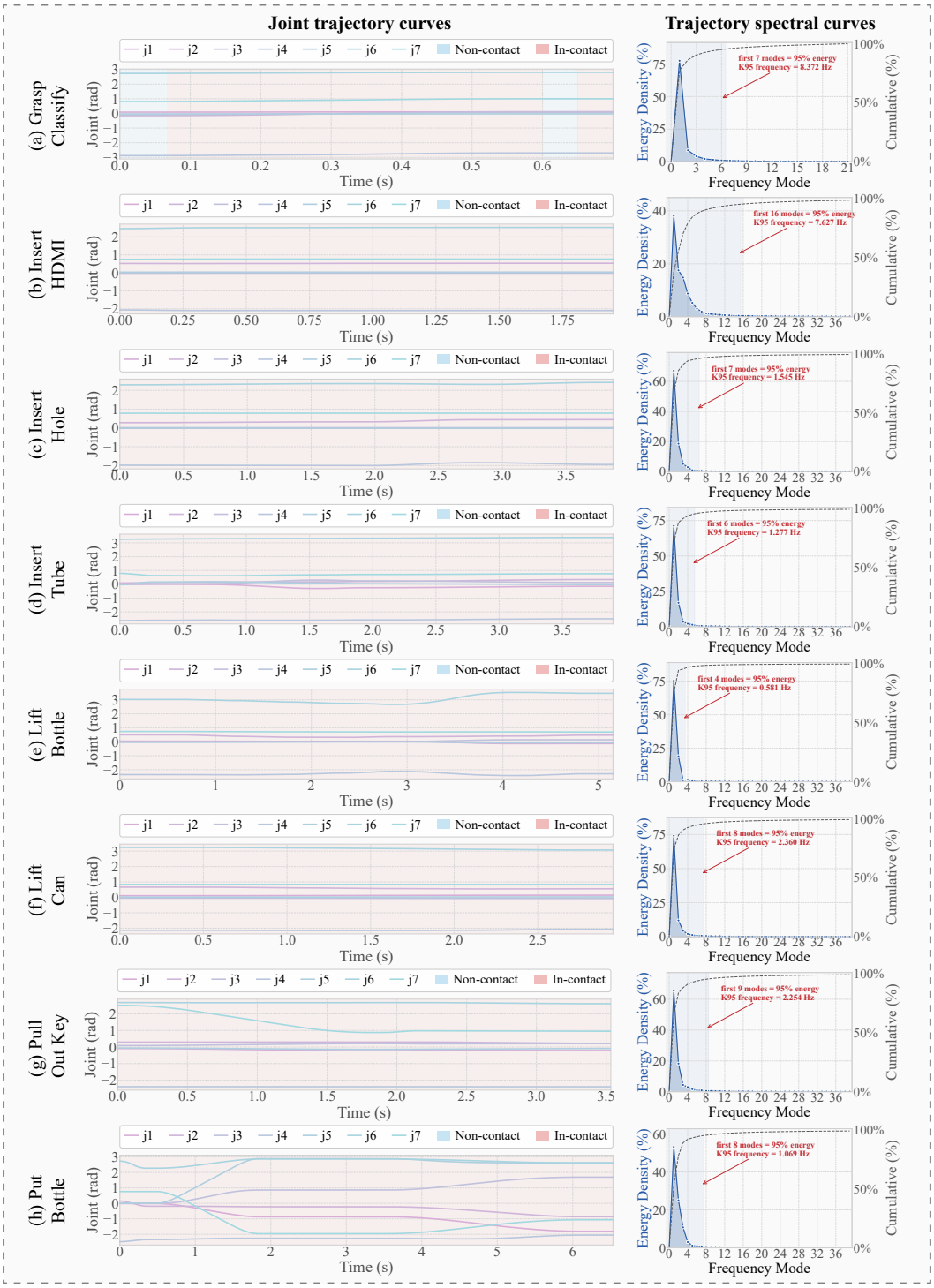


Figure S4: Joint-space trajectories and their mean-centered FFT spectra for eight UniVTAC tasks, using one complete trajectory per task. (a) Grasp Classify. (b) Insert HDMI. (c) Insert Hole. (d) Insert Tube. (e) Lift Bottle. (f) Lift Can. (g) Pull Out Key. (h) Put Bottle in Shelf.

predicts action chunks from the fused representation. Code is available at https://github.com/Shengqi77/ViTacMotor.

(3) Visuo-Tactile-Language Action Methods.

TacVLA (Zhang et al., 2026c) incorporates tactile inputs into a pretrained vision-language-action policy. For our experiments, we implement a TacVLA baseline on the $\pi_{0.5}$ framework following the method description in the paper, incorporating tactile inputs alongside visual observations, language instructions, and proprioception for action generation.

ForceVLA (Yu et al., 2025) is a force-aware vision-language-action model that leverages tactile feedback to enhance the action generation process. It employs a multi-modal approach to integrate visual, tactile, and language information for improved manipulation performance. Note that in our experiments, we replace force feedback with tactile feedback to align with the tactile sensing setup. The code is available at https://github.com/ft-robotic/ForceVLA/.

Tactile-VLA (Huang et al., 2025b) is initialized from a pretrained π_0 model and performs token-level fusion of visual observations, language instructions, proprioception, and a temporal history of tactile measurements. Its tactile-aware action expert is trained with conditional flow matching to predict both a target robot pose and a target contact force. A hybrid position-force controller converts the discrepancy between the target and measured forces into a corrective position command, enabling language-conditioned regulation of interaction strength. The paper additionally presents Tactile-VLA-CoT, which periodically reasons over tactile feedback and generates corrective instructions after detecting task failures. The code is not publicly available, so we only compare against this method in simulation and directly use the experimental comparison data reported in FTP-1 (Yuan et al., 2026).

FTP-1 (Yuan et al., 2026) is a tactile foundation policy designed to transfer across heterogeneous sensors and robot embodiments. It maps image-based, array-based, and state-based tactile signals into a Morphology-Aware Tactile Token Space using sensor-specific encoders and shared functional-area embeddings. Built on $\pi_{0.5}$, FTP-1 employs separate vision-language, tactile, and action transformer experts, where the flow-matching action expert attends to the shared tactile expert without modifying the pretrained vision-language representations. The model is pretrained on approximately 3,000 hours of tactile manipulation data collected from 26 data sources and 21 tactile sensors before task-specific fine-tuning. We only compare against this method in simulation and directly use the experimental comparison data reported in FTP-1 (Yuan et al., 2026). The code is available at https://github.com/michaelyuancb/ftp1-policy.

B Method Details and Discussion

B.1 Efficient Optical Flow Algorithm for Tactile Motion Estimation

We use DIS optical flow (Kroeger et al., 2016) to estimate pixel motion between tactile images. The algorithm combines inverse search, a multi-scale pyramid, and variational refinement. Table S2 summarizes the parameter settings.

B.2 Spectral Analysis of Joint-Space Trajectories across Additional Tasks

Figures S3 and S4 show one complete trajectory per task for four additional real-world tasks and eight UniVTAC tasks, respectively. We analyze the arm-joint channels, excluding the grippers, using joint commands for ALOHA and UniVTAC and measured joint positions for Flexiv. For each channel, we subtract its temporal mean and apply a real-input FFT (rFFT). We sum the squared coefficient magnitudes across channels and normalize over the returned frequency bins. K_{95} is the minimum number of consecutive bins, starting at zero frequency, that account for at least 95% of this sum. Bin k corresponds to $f_k = kf_s/T$, where f_s is the sampling rate and T is the number of frames. Shading indicates estimated contact intervals for visualization; the spectra are computed over the full trajectories, not separately for each interval.

These results show low-frequency dominance in the evaluated trajectories, supporting our choice of refinement subspace. This FFT-based analysis characterizes recorded trajectories, whereas the policy uses DCT for action-chunk refinement; their mode counts are not directly interchangeable.

B.3 Shared Tactile Guidance and Slow Expert

Guidance Encoder. Following the two-branch structure in Eq. 4, each motion channel is encoded by a linear layer with a SiLU activation and 64 output features. The two encoders have independent

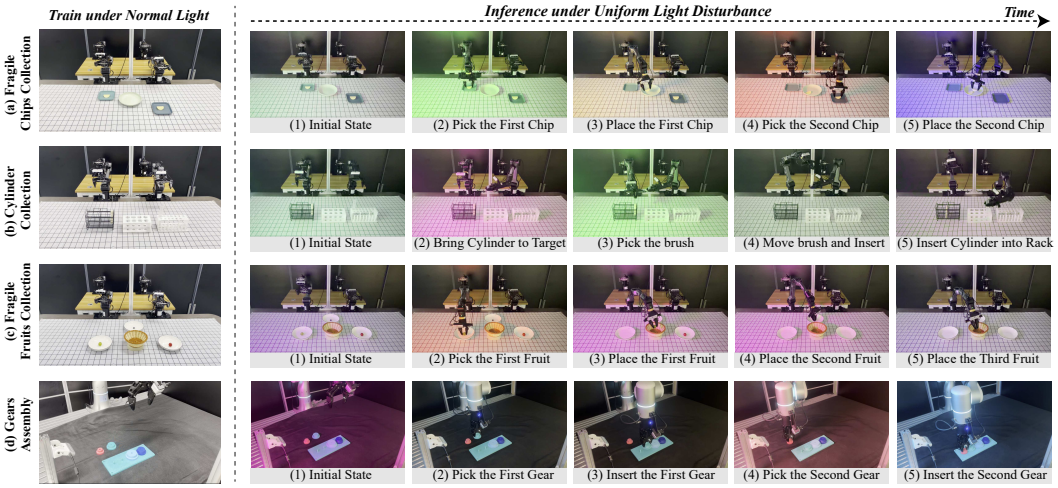


Figure S5: Representative successful executions under uniform lighting disturbances: (a) Chips Collection. (b) Cylinder Collection. (c) Fruits Collection. (d) Gears Assembly.

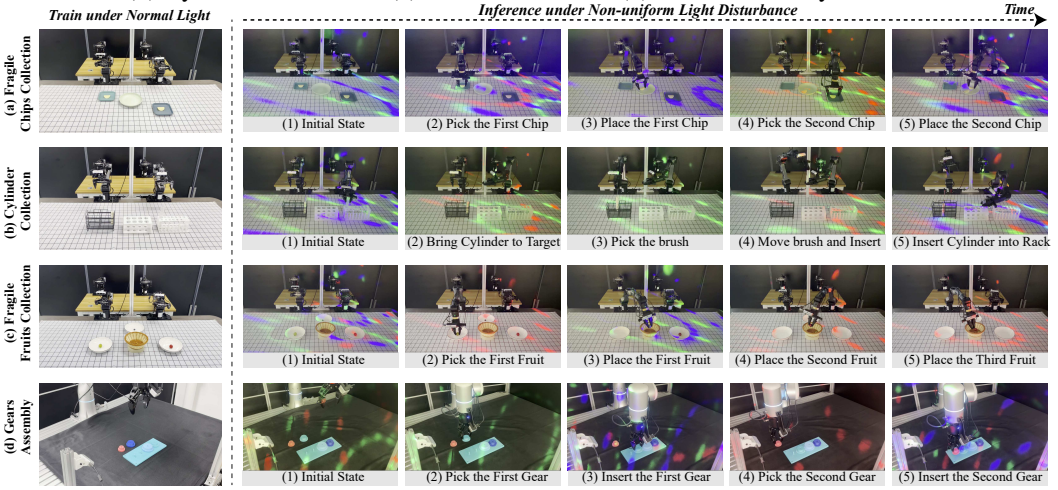


Figure S6: Representative successful executions under non-uniform lighting disturbances: (a) Chips Collection. (b) Cylinder Collection. (c) Fruits Collection. (d) Gears Assembly.

parameters; sharing refers to the fused guidance used by the slow and fast experts, not to weight sharing between the encoders. The concatenated features are projected by the fusion layer $\mathcal{F}$ to a 64-dimensional guidance vector $\mathbf{T}_t^{\text{pro}}$. In Eq. 3, the transient descriptor is the signed difference between consecutive cumulative motion magnitudes. A positive value indicates increasing deformation magnitude, whereas a negative value indicates decreasing magnitude. Retaining this sign is part of the descriptor definition in Section 3.1: taking an absolute value or clipping negative values to zero would remove this directional information. The two branches in Eq. 4 therefore represent complementary information about deformation magnitude and its temporal change.

Routing Implementation. The router R in Section 3.2 is a linear projection to $K = 4$ logits. In the implementation, the logits are first clipped element-wise to $[-6, 6]$, then divided by a temperature $\tau = 1.0$, and finally normalized with softmax:

$$\mathbf{w}_t = \text{softmax}\left(\frac{\text{clip}(R(\mathbf{T}_t^{\text{pro}}), -6, 6)}{\tau}\right), \qquad \tau = 1.0. \tag{12}$$

This makes explicit the logit clipping omitted from the compact routing expression in Section 3.2. A single router computes the weights for the current observation. The same weights are broadcast across action tokens and reused in all tactile-aware MoE layers; there is no independent router in each transformer layer. The shared and stage-specific branches are feed-forward modules within the action transformer. The branch combination is defined in Section 3.2. Routing is learned through the action objectives without manual stage labels. Training settings are summarized in Table S1.

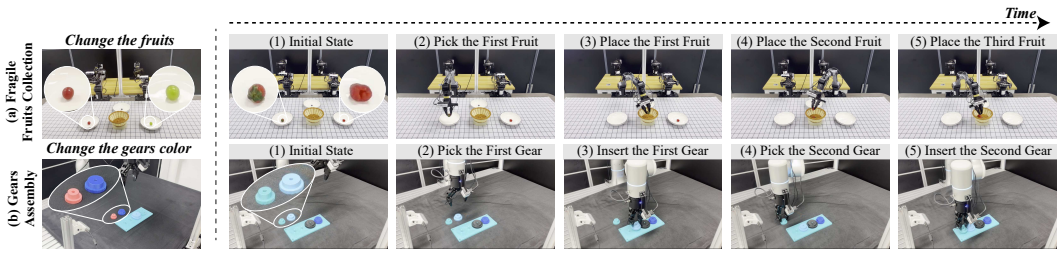


Figure S7: Representative successful executions with object variations. We replace grapes and tomatoes with raspberries and strawberries in the fruits collection task and change the gear colors in the gears assembly task.

B.4 Fast Expert Formulation

Spatial Encoding. For the fast expert in Section 3.3, the real-world spatial encoder E_{s} is a ResNet-18 with a two-channel input layer and GroupNorm. It maps each normalized 64×64 cumulative flow field to a 512-dimensional feature.

Temporal Encoding. Following Eq. 7, the spatial encoder E_{s} extracts features from the latest $N = 5$ cumulative motion fields, and TAgg combines them into the window-level embedding $\mathbf{z}_t$. The GRU updates $\mathbf{T}_t^{\mathrm{rea}}$ from $\mathbf{z}_t$ and the previous hidden state $\mathbf{T}_{t-1}^{\mathrm{rea}}$. In this formulation, the recurrent state carries contact history beyond the explicit window, while all inputs remain causal.

Residual Parameterization. Equation 9 defines G as a coefficient-space corrector conditioned on $[\mathbf{T}_t^{\mathrm{rea}}; \mathbf{p}_t]$. Its specified output is $\Delta\mathbf{c}_t \in \mathbb{R}^{M \times d_a}$, containing one correction per retained mode and action dimension. The corresponding action-space update in Eq. 11 can be expressed as

$$\hat{\mathbf{A}}_t - \mathrm{sg}(\mathbf{A}_t^{\mathrm{base}}) = g_t^R \Phi_M \Delta\mathbf{c}_t, \qquad \Phi_M = [\boldsymbol{\phi}_0, \ldots, \boldsymbol{\phi}_{M-1}], \tag{13}$$

where $\Phi_M \in \mathbb{R}^{H_a \times M}$ contains the inverse-DCT basis vectors for modes $0, \ldots, M-1$, including the constant mode, and g_t^R is the scalar contact gate defined in Eq. 10. Thus, the formulation restricts the added correction to an M-mode temporal subspace, while leaving higher-order coefficients of the base prediction unchanged.

Online Execution. The slow and fast update rates are configured as 1 Hz and 25 Hz, respectively. The slow expert refreshes a cached base action chunk, while the fast expert uses incoming tactile observations to refine the entire cached chunk. Here, $\mathbf{A}_t^{\mathrm{base}}$ denotes the latest available base chunk at time t. Each refinement is applied to this base chunk, rather than to the previously refined output, so corrections do not accumulate across updates. Execution advances through the refined chunk according to elapsed time; fast updates do not restart this progression. When a new base chunk becomes available, it replaces the cached chunk and the execution timing is reset for the new chunk. The effective update rates depend on inference and communication latency.

B.5 Rethinking the Role of Touch through Policy Design Comparisons

GuidedVTLA uses shared tactile guidance to coordinate stage-aware action generation and low-frequency refinement. The following comparisons clarify this design along four complementary dimensions.

Action and Task Structure. Visual imitation policies learn observation-to-action mappings (Cheng et al., 2025; Ge et al., 2025; Ni et al., 2025; Tang et al., 2025; Wang et al., 2024; Ze et al., 2024; 2025), with action chunking and diffusion providing temporal structure and multimodal action modeling (Zhao et al., 2023; 2024; Chi et al., 2025). VLA models further incorporate language-conditioned task information (Brohan et al., 2022; Intelligence et al., 2025; Kim et al., 2025; 2024; Liu et al., 2024). Task structure can be supplied through separate planning modules (Zhou et al., 2024; Yang et al., 2025; Shi et al., 2025; Huang et al., 2024b; Erdogan et al., 2025; Team et al., 2025) or closer integration of reasoning and action generation (Chen et al., 2025a; Fang et al., 2025; Zhang et al., 2026d; Lin et al., 2025; Liu et al., 2025c). These choices concern how a task is organized; tactile dynamics provide a complementary cue about how physical interaction evolves during execution. GuidedVTLA uses this cue for stage-aware routing without requiring manual contact-stage labels.

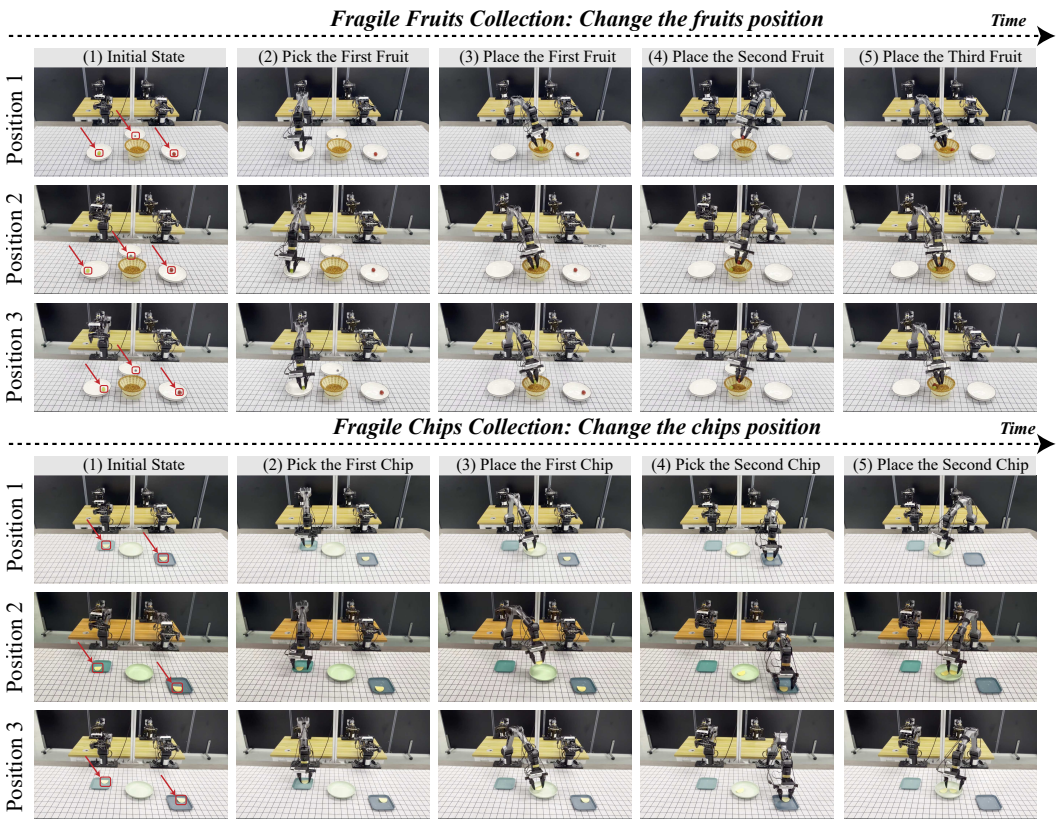


Figure S8: Representative successful executions from varied initial object positions on the dual-arm platform. We randomize the initial placements of target objects in the fruits collection and chips collection tasks. **The red arrows and bounding boxes** indicate the positions of the objects.

Where Touch Enters the Policy. Visuo-tactile policies can combine visual and tactile features (Chen et al., 2022; Li et al., 2022), jointly process the modalities (Li et al., 2025b), or adapt their weighting through force-related attention (Li et al., 2025a). Within language-conditioned policies, touch can enter through multimodal representations (Huang et al., 2025b; Liu et al., 2025d), action-generation modules (Li et al., 2026; Yu et al., 2025), residual corrections (Zhang et al., 2026e), or textual prompts (Van der Merwe et al., 2026). Thus, where touch enters a policy and what role it plays are distinct design choices. GuidedVTLA connects two intervention points: shared tactile guidance conditions routing in the slow expert and conditions low-frequency refinement in the fast expert, giving base generation and subsequent correction a common tactile context.

What Touch Contributes. Force-aware policies and tactile-force-controlled grippers use contact measurements to adjust physical interaction (He et al., 2025; Kang et al., 2026). Other approaches exploit predicted tactile observations (Ye et al., 2025), feedback at multiple levels (Bi et al., 2025), or preference learning for insertion (Zhang et al., 2025). These examples show that tactile reasoning extends beyond instantaneous reactive feedback. The distinction of GuidedVTLA is therefore not simply the use of touch for anticipation or correction. Rather, the same temporal tactile guidance coordinates stage-aware action generation with spectral refinement: the former adapts the action-generation strategy, while the latter adjusts the retained motion plan using incoming tactile feedback.

Representation and Data. These policy-level choices assume that useful tactile information is available. Visuo-tactile pretraining (George et al., 2025), simulation-based assembly learning with visuo-tactile adaptation (Huang et al., 2025a), and robot-free demonstration collection (Liu et al., 2025a) address how useful contact information is learned or acquired. These efforts are complementary to the policy-level choices above: GuidedVTLA concerns how available tactile information coordinates base action generation and subsequent refinement, rather than prescribing a new data-collection interface.

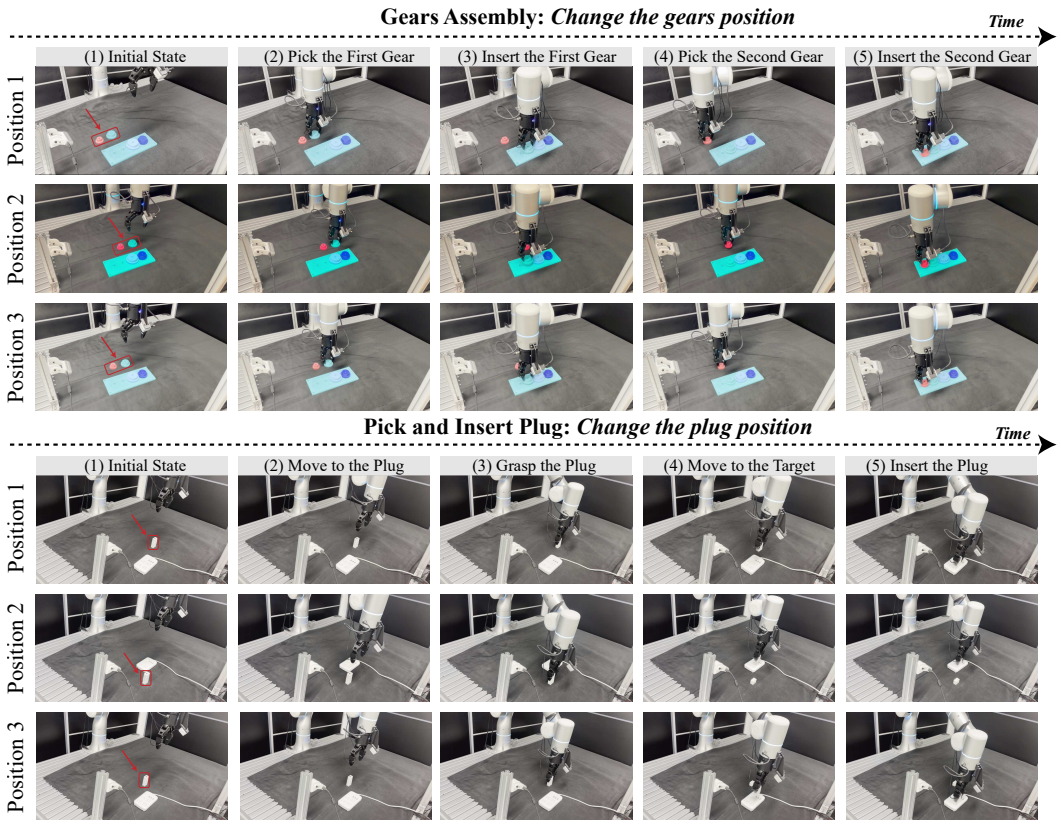


Figure S9: Representative successful executions from varied initial object positions on the single-arm platform. We randomize the initial placements of target objects in the gears assembly and plug insertion tasks. **The red arrows and bounding boxes** indicate the positions of the objects.

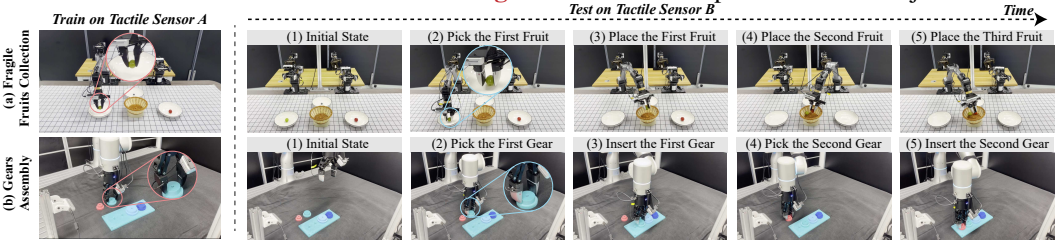


Figure S10: A representative successful zero-shot execution on Sensor B (Dai) using a model trained with Sensor A (Neo). The model successfully completes the task in the illustrated execution.

C Analysis and Discussion

C.1 Robustness to Environment and Object Variations

We present qualitative results under lighting changes and object variations. For lighting changes, Figures S5 and S6 show representative successful executions under uniform and non-uniform lighting disturbances. Our method successfully completes the manipulation tasks in the illustrated cases. For object variations, we replace grapes and tomatoes with raspberries and strawberries in the fruits collection task and change the gear colors in the gears assembly task. Figure S7 shows that our method successfully completes the tasks with these object variations in the illustrated cases.

C.2 Generalization to Object Position Variations

We vary the initial object positions on both the bimanual and single-arm platforms. Figure S8 shows representative successful executions of fruits collection and chips collection on the bimanual platform. Figure S9 shows successful executions of gears assembly and plug insertion on the single-arm

platform. Our method successfully completes the tasks from the varied initial positions shown in these examples.

C.3 Choice of Representation for Proactive Guidance and Reactive Refinement

We compare tactile dynamics with raw tactile images as inputs for proactive guidance and reactive refinement. Figure 7 reports higher success rates for tactile dynamics in both within-sensor evaluation on Sensor A and zero-shot transfer from Sensor A to Sensor B. The observed decrease in success rate from within-sensor evaluation to cross-sensor transfer is also smaller for tactile dynamics than for raw tactile images. These results support the use of tactile dynamics for cross-sensor transfer in the evaluated tasks and sensor pair.

C.4 Cross-sensor Generalization of the Proposed Method

Figure S10 presents a representative successful zero-shot execution on Sensor B (Dai) using a model trained with Sensor A (Neo). The model successfully completes the task in the illustrated execution without additional policy training on Sensor B.

C.5 Limitation and Future Work

Although GuidedVTLA achieves promising performance on multi-stage contact-rich manipulation tasks, it still lacks the capacity for autonomous adjustment and recovery when subjected to severe human disturbances or large spatial pose variations. In future work, we plan to address these limitations and investigate whether tactile feedback can contribute to improved action recovery.