

FTP-1: A Generalist Foundation Tactile Policy Across Tactile Sensors for Contact-Rich Manipulation

Chengbo Yuan^{‡,*}, Zicheng Zhang^{*,3,7}, Mingjie Zhou^{*,3,7}, Wendi Chen^{4,8}, Yi Wang^{4,8}, Zhuoyang Liu⁵,
Dantong Niu⁵, Shuo Wang¹, Hui Zhang⁶, Wenkang Zhang¹, Yingdong Hu¹, Yuanqing Gong³,
Wanli Xing³, Chuan Wen⁴, Cewu Lu^{4,8}, Kaifeng Zhang³, Yang Gao^{†,1,2}

¹Tsinghua University ²Shanghai Qi Zhi Institute ³Sharpa
⁴Shanghai Jiao Tong University ⁵University of California, Berkeley
⁶ETH Zurich ⁷Fudan University ⁸Shanghai Innovation Institute

*Equal contribution. ‡Project Leader †Corresponding author

Abstract: Despite the success of vision-based generalist robotic policies, existing tactile-based policies remain tied to fixed embodiments and sensor setups. This is because tactile signals are highly heterogeneous across hardware, making cross-sensor generalization difficult. We present FTP-1, **the first generalist foundation tactile policy pretrained to acquire transferable tactile manipulation abilities** across diverse sensors and embodiments. FTP-1 supports varied tactile inputs, including image-, array-, and state-based signals, by using heterogeneous encoders to project them into unified morphology-aware latent tokens that are jointly modeled by a shared tactile Transformer expert. Pretrained on $\sim 3,000$ hours of tactile manipulation data aggregated from 26 data sources, spanning human and robot demonstrations across 21 sensors, FTP-1 learns tactile skills that transfer beyond the sensors seen during pretraining. Across downstream finetuning experiments spanning 5 hardware configurations, FTP-1 improves contact-rich manipulation on seen sensor setups by +17.2% and, surprisingly, transfers to two previously unseen tactile-sensor setups, achieving a +31% gain in success rate. FTP-1 establishes the first unified foundation baseline for tactile manipulation, providing future tactile policies with a shared model-level starting point. Pretrained models, datasets, training code and more visualization at <https://ftp1-policy.github.io/>.

Keywords: Tactile Manipulation, Generalist Policy, Contact-Rich Tasks.

1 Introduction

Generalist robotic policies [1, 2] seek to move robot learning beyond task-specific training [3, 4] toward large-scale pretrained policy models that can be efficiently adapted to diverse embodiments, tasks, and environments. By scaling data, computation, and model capacity over large-scale heterogeneous datasets, recent vision-based generalist policies, such as $\pi_{0.5}$ [5], have shown that pretraining provides powerful initialization for downstream manipulation tasks.

However, this generalist paradigm remains largely underexplored in tactile-based policy learning. This is a critical gap, as tactile sensing plays an essential role in contact-rich, fine-grained manipulation. Recent pretrained tactile policies [6, 7, 8] have begun to scale tactile data for policy pretraining, but remain tied to limited sensor configurations, observation formats, and robot embodiments, rather than a generalist policy. This sensor-specific design is fundamentally constrained by the heterogeneity of tactile signals across hardware, where sensors differ in modality, resolution, morphology, and contact response. In this work, we ask whether tactile manipulation [9, 10] can be learned under a generalist foundation policy paradigm [11]: can a single tactile policy absorb heterogeneous tactile experience and transfer to sensors and embodiments beyond those seen during pretraining?

We present **FTP-1, the first generalist Foundation Tactile Policy pretrained to acquire transferable tactile manipulation abilities across diverse sensors and embodiments** (Fig. 1). FTP-1 supports diverse tactile inputs, including image-, array-, and state-based signals. To address tactile

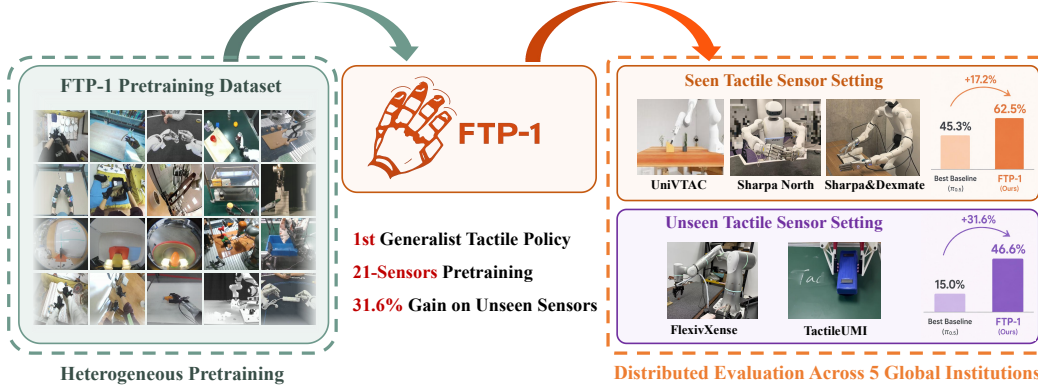


Figure 1: We present FTP-1, the first generalist foundation tactile policy pretrained for diverse sensors and embodiments. Pretrained on a large-scale heterogeneous tactile manipulation dataset, FTP-1 improves downstream performance with a 31.6% point gain on unseen sensor setups.

heterogeneity, we introduce Morphology-Aware Tactile Token Space (MTTS) as a unified interface across sensors. MTTS represents tactile inputs with a fixed set of functional-area tokens, each augmented by a shared function-area embedding. Heterogeneous tactile encoders are then used to project different tactile inputs into this common space. On top of MTTS, FTP-1 adopts a multi-expert foundation-policy architecture [1, 5], where a shared tactile Transformer expert [12] jointly models tactile tokens to learn reusable tactile representations and manipulation skills across sensors.

We further curate FTP-1-Dataset, a new large-scale heterogeneous tactile manipulation dataset aggregated from 26 data sources. It contains approximately 3,000 hours of human and robot demonstrations across 21 tactile sensors, with all data standardized through our unified MTTS tactile interface. Pretraining on this dataset enables FTP-1 to acquire transferable tactile manipulation skills across sensors and embodiments.

To evaluate this capability, we distribute FTP-1 pretrained checkpoints to 5 independent institutions spanning global regions, each conducting finetuning on a distinct hardware setup, covering 4 tactile sensors in total. The task suite comprises 14 diverse tasks, covering a broad range of contact-rich behaviors, including in-hand adjustment, force-controlled pressing, insertion and extraction, and long-horizon fine-grained dexterous manipulation. Results show that FTP-1 improves success rates by +17.2% on 3 seen sensor setups and, surprisingly, also transfers to previously 2 unseen tactile sensors, achieving a +31% success-rate gain. Ablation studies further confirm that these gains stem from FTP-1’s transferable tactile manipulation skills. In summary, our contributions are:

- **FTP-1, the first generalist foundation tactile policy** with transferable tactile manipulation skills across diverse sensors and embodiments, through morphology-aware tactile tokenization and shared tactile expert modeling.
- **A large-scale aggregated tactile manipulation dataset** aggregated from 26 data sources, spanning human and robot data and 21 distinct tactile sensors. By pretraining on this, FTP-1 establishes a shared foundation starting point for tactile manipulation policy training.
- **Results of tactile manipulation knowledge transfer.** Experiments show that FTP-1 improves downstream contact-rich manipulation, even transfer to unseen tactile-sensor setups with 31% success rate improvement.

2 FTP-1: Generalist Foundation Tactile Policy

We introduce FTP-1, the first generalist tactile foundation policy for contact-rich manipulation across diverse tactile sensors and robot embodiments. Given a language instruction ℓ , (multi-view) RGB observations $\mathcal{I}_t$, proprioception $\mathbf{s}_t$, and tactile observations $\mathcal{X}_t$, FTP-1 predicts an action chunk $\hat{\mathbf{A}}_{t:t+H-1} = \pi_\theta(\ell, \mathcal{I}_t, \mathbf{s}_t, \mathcal{X}_t)$, where $\hat{\mathbf{A}}_{t:t+H-1} \in \mathbb{R}^{H \times D}$, H is the action horizon, and D is the dimension of a predefined Unified Action Space (UAS) [13] that handles heterogeneity of control

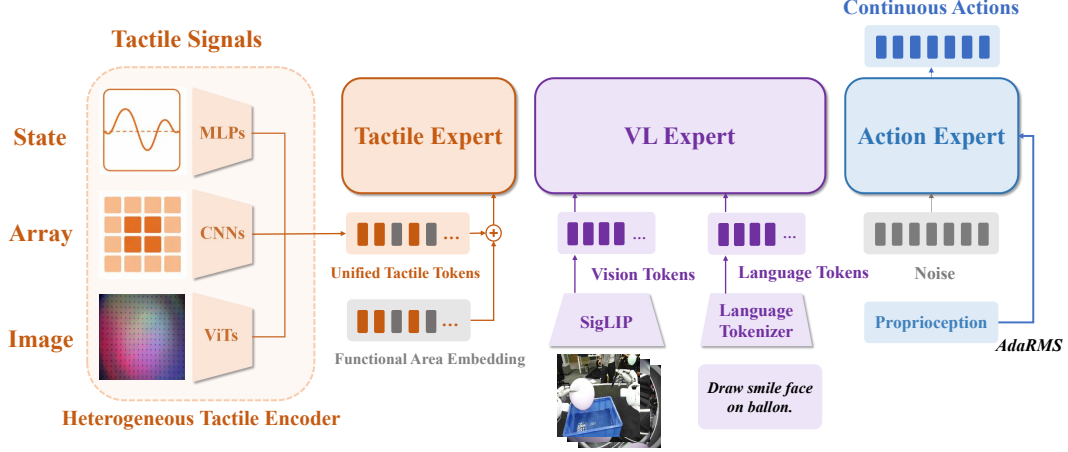


Figure 2: Overview of FTP-1 architecture. Heterogeneous tactile observations are mapped into a unified tactile token space through sensor-specific encoders, processed by a shared tactile expert, and fused with vision-language and proprioceptive information for action generation.

signal, which is detailed in App. B.1. We next describe FTP-1’s key components: the Morphology-Aware Tactile Token Space (MTTS) definition as the unified tactile interface (Sec. 2.1), heterogeneous tactile encoders for MTTS tokenization (Sec. 2.2), a shared tactile expert for modality fusion (Sec. 2.3), and the pretraining dataset (Sec. 2.4).

2.1 Morphology-Aware Tactile Token Space (MTTS)

Tactile sensors vary in modality, resolution, and morphology, making cross-sensor representation sharing challenging. FTP-1 addresses this challenge with a Morphology-Aware Tactile Token Space (MTTS), augmented by shared functional-area embeddings. MTTS organizes tactile signals into 24 functional areas [14], each represented by one token; the full definition is shown in Fig. 3. For in-hand tactile signals, we use slots 0-14 to represent different hand functional regions. For force/torque signals from wrists and fingers, we use slots 15-20. For parallel grippers, the two gripper-side sensors are mapped to the thumb-tip slot (slot 0) and index-fingertip slot (slot 1), reflecting their two-finger grasping function. Slots 21-23 are reserved for future use.

Under this definition, sensor-specific tactile signals are grouped by functional area and encoded into unified morphology-aware tactile tokens. To indicate the functional area of each token, we add a learnable functional-area embedding, shared across all sensors, before feeding the tokens into the subsequent architecture. We use separate functional-area embeddings for the left/right hands to distinguish their tactile tokens.

2.2 Heterogeneous Tactile Encoders for MTTS Tokenization

Although MTTS provides a unified token interface, tactile inputs from different sensors still vary in shape and modality. We therefore use heterogeneous tactile encoders [15] to tokenize sensor-specific inputs into MTTS. For each sensor, we first group its signals according to MTTS functional areas, then categorize each group into one of three observation types: image, array, or state.

For image-type inputs (e.g., GelSight [16]), we use a lightweight sensor-specific ViT [17] followed by a shared pretrained T3 Transformer tactile encoder [18] across sensors, and use the final [CLS] token as the tactile token. For array-type inputs (e.g., Contactile [19]), we use a CNN [20] to capture spatial tactile structure and compress each functional area into one token. For state-type inputs



Figure 3: Tactile functional-area definition of MTTS.

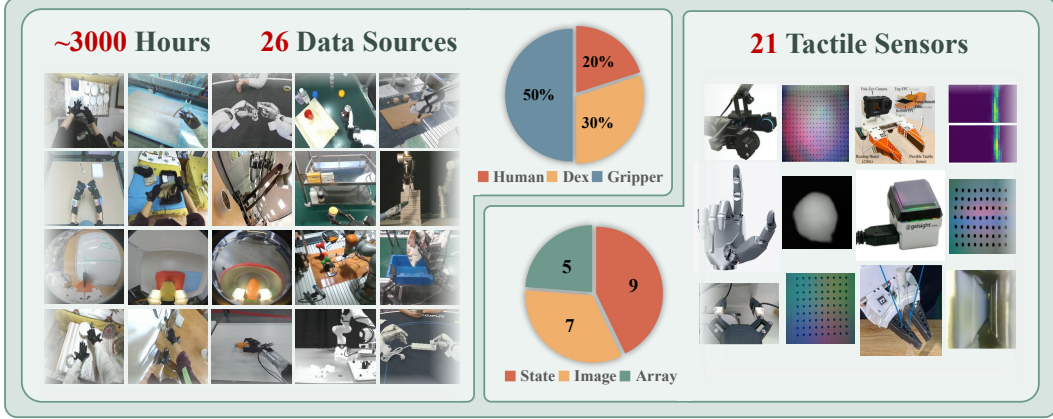


Figure 4: Overview of the FTP-1-Dataset. The dataset aggregates 26 sources across human and robot manipulation, covering 21 tactile sensors with image-, array-, and state-based modalities, all organized under the MTTS interface for unified pretraining.

(e.g., force-torque), we Fourier-encode the raw state [21] and process it with a lightweight MLP. For sensors with multiple functional areas of the same observation shape, we share the corresponding encoder across these areas to reduce sensor-specific parameters and encourage common tactile dynamics to be modeled consistently. Details are provided in App. B.2.

2.3 Shared Tactile Expert for Modality Fusion

Finally, FTP-1 adopts a multi-expert architecture [1] to fuse tactile tokens with vision-language perception for action generation, as shown in Fig. 2. Built on $\pi_{0.5}$ [5], FTP-1 processes image observations and language instructions with a pretrained vision-language Transformer expert [22], whose outputs are attended by a flow-matching Transformer action expert [1]. We further find that fusing proprioceptive state with adaptive RMSNorm [5] improves performance and generalization, as detailed in App. B.3.

Unlike prior tactile-augmented VLA models that inject tactile inputs into the vision-language expert via lightweight adapters [23, 6, 9, 24], FTP-1 uses an independent tactile expert for extracted tactile tokens. This design (1) supports reuse pretrained shared tactile expert on unseen sensors during finetuning (Section 4.1), promotes transferable tactile manipulation skills, and (2) avoids disturbing pretrained vision-language knowledge. It also (3) improves tactile processing efficiency and contact-rich manipulation performance, as shown by our comparison with adapter-based methods such as Tactile-VLA [9] (Sec. 3 and Sec. 4.1). The action expert attends to the tactile expert, but not vice versa. The tactile expert is chosen as a 300M-parameter Transformer [12]. More complex modality fusion designs e.g. MoE-based fusion [25, 24, 26], did not yield consistent gains in our experiment, so we adopt the simplest multi-expert design. Together, these choices provide a simple modular architecture for learning transferable tactile manipulation skills across sensors and embodiments.

2.4 FTP-1 Pretraining Datasets

FTP-1 is pretrained on a large-scale heterogeneous tactile manipulation dataset (Fig. 4) aggregated from 26 sources, covering 21 distinct tactile sensors: 7 image-type, 5 array-type, and 9 state-type sensors. In addition to aggregating existing datasets, we collect Sharpa North-FTP-1 with 4,000 long-horizon dexterous demonstrations. All tactile annotations are organized according to MTTS functional areas, described in Sec. 2.1. To mitigate data imbalance, we apply dataset-specific sampling ratios during training; after re-sampling, the proportions of human, dexterous-hand robot, and gripper robot data are approximately 20%, 30%, and 50%, respectively. The final aggregated dataset contains around 3,000 hours of tactile manipulation data. Details are provided in App. C.

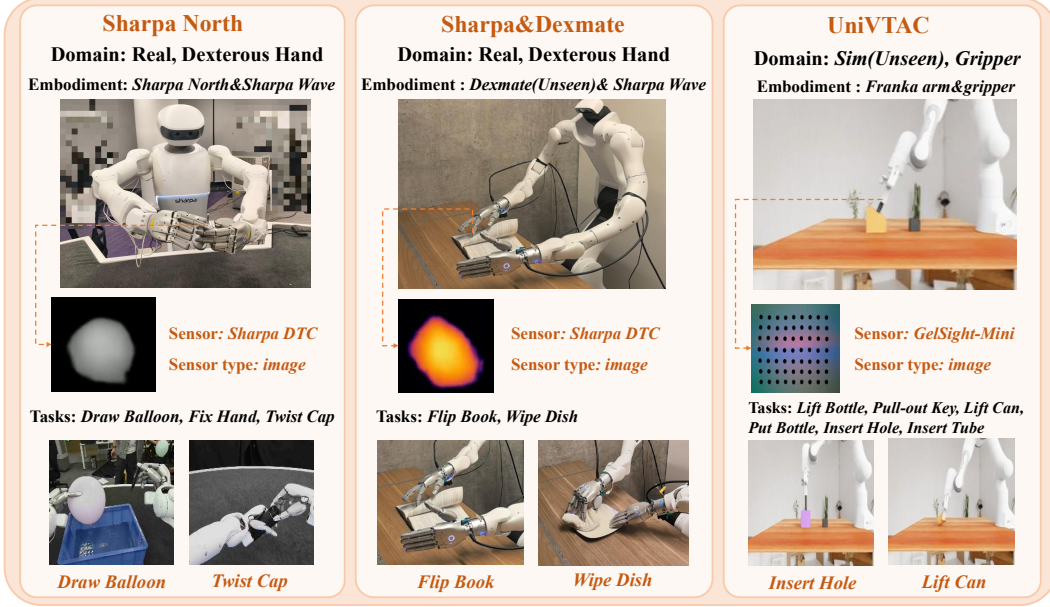


Figure 5: Overview of the finetuning experiment for seen sensors during pretraining.

3 Finetuning Experiment

In this section, we address the central question: **Does FTP-1 pretraining improve downstream tactile policy finetuning by learning effective representations for tactile sensor perception?** To answer this, we distribute FTP-1 pretrained checkpoint to 3 independent institutions, each evaluates on a distinct embodiment and task suite across simulation and real-robot settings. Training details are provided in App. D. We next describe the evaluation setup and results.

3.1 Baselines and Metrics

For simulation experiments, we evaluate 100 rollouts per task, following the original benchmark [27]. For real-robot experiments, we evaluate 20 rollouts per task. We report finetuned success rates and compare the pretrained FTP-1 with three baselines:

- $\pi_{0.5}$ [5]: a SOTA open-source VLA model without tactile input. This baseline evaluates the benefit of tactile sensing over a strong vision-language-action policy.
- **Tactile-VLA** [9]: a tactile-based VLA architecture adapted from Huang et al. [9], which injects tokenized tactile inputs into the VLM expert without a separate tactile expert. This baseline evaluates the effect of our tactile-expert design.
- **FTP- $\pi_{0.5}$** : our architecture partially initialized with $\pi_{0.5}$ weights but without FTP-1 pretraining. This baseline isolates the contribution of large-scale tactile pretraining.

These baselines are chosen to disentangle the effects of tactile input, tactile integration architecture, and FTP-1 pretraining. All methods are implemented based on the $\pi_{0.5}$ [5] codebase with comparable model scale, finetuning data, and training protocol for a fair comparison.

3.2 Task Settings

We evaluate FTP-1 on three embodiments: UniVTAC [27] in simulation, and Sharpa North and Sharpa&Dexmate in the real world. An overview of the evaluation setups is shown in Fig. 5. These setups cover two tactile sensors included in pretraining, GelSight-Mini [16] and Sharpa DTC [10], so both the tactile tokenizer and tactile expert of our methods are initialized from FTP-1 checkpoints. For UniVTAC, we choose six contact-rich tasks spanning in-hand manipulation and contact-aware insertion/extraction; Sharpa North focuses on three long-horizon dexterous manipulation tasks: Draw Balloon, Fix Hand and Twist Cap, including deformable-object interaction, small-part as-

Table 1: Results on the UniVTAC simulation benchmark. We report success rates (%). “Avg. w/o Lifts” excludes Lift Bottle and Lift Can. The best and second-best results are highlighted in **bold** and underline, respectively. Results marked with * are from the original benchmark [27].

Method	Lift Bottle	Pull-out Key	Lift Can	Put Bottle	Insert Hole	Insert Tube	Avg.	Avg. w/o Lift
VITaL* [28]	72	<u>47</u>	8	32	25	34	36.33	34.5
UniVTAC-ACT* [27]	71	46	29	31	25	56	43.00	39.5
$\pi_{0.5}$ [5]	97	38	72	16	31	41	<u>49.16</u>	31.5
Tactile-VLA [9]	97	32	15	10	41	56	41.83	34.75
FTP- $\pi_{0.5}$	<u>77</u>	30	26	<u>19</u>	<u>47</u>	<u>72</u>	45.16	<u>42</u>
FTP-1	97	48	<u>65</u>	47	64	79	66.66	59.5

Table 2: Real-robot results on seen tactile sensor setups. We report finetuned success rates (%). The best and second-best results are highlighted in **bold** and underline, respectively.

Method	Sharpa North				Sharpa&Dexmate		Average
	Draw Balloon	Fix Hand (Tear)	Fix Hand (Finish)	Twist Cap	Flip Book	Wipe Dish	
$\pi_{0.5}$ [5]	35	70	35	40	65	30	45.3
Tactile-VLA [9]	20	80	25	10	45	35	35.8
FTP- $\pi_{0.5}$	25	65	25	20	<u>70</u>	<u>45</u>	41.6
FTP-1	45	80	40	65	85	60	62.5

sembly, and bimanual twisting behaviours. Sharpa&Dexmate focuses on pressing and force-control behaviors, including Flip Book and Wipe Dish tasks. Hardware setups, task definitions, experiment details, and excluded UniVTAC tasks are provided in App. E.1 and App. E.2. Examples of policy rollouts are provided in App. E.3.

3.3 Experiment Results

Simulation Results. Tab. 1 reports the UniVTAC benchmark results [27]. Surprisingly, Lift Bottle and Lift Can can be largely solved without tactile input in simulation: $\pi_{0.5}$ already achieves 97% and 72% success rates, respectively. We therefore additionally report the average success rate excluding these two tasks. FTP-1 achieves the best performance under both metrics, with 66.66% overall success and 59.5% excluding the lift tasks, outperforming the second-best method by about +17.5% in both cases.

Real-World Results. Results are shown in Tab. 2. FTP-1 achieves the best performance among all methods, with an average success rate of 62.5%. Surprisingly, $\pi_{0.5}$ ranks second with a 45.3% average success rate, outperforming the other two tactile-based baselines. This suggests that, without proper modality fusion, tactile inputs may hurt performance by interfering with the vision-language perception module, consistent with observations in previous work [14], especially on long-horizon tasks in the Sharpa North domain.

In terms of detailed behavior, Tactile-VLA [9] and FTP- $\pi_{0.5}$ tends to generate unstable actions when contact conditions change, indicating limited robustness to tactile perception without our tactile-expert fusion design. Without tactile inputs, $\pi_{0.5}$ fails to maintain consistent pressing force on the two Sharpa&Dexmate tasks sometimes and pushes against the bottle cap without reactive force adjustment. In contrast, FTP-1 produces more stable and smoother actions across these tasks.

4 Experiment of Transferable Tactile Manipulation Abilities

While prior work shows that in-domain tactile-sensor pretraining improves downstream policy performance [18, 6, 8], whether large-scale heterogeneous pretraining enables transfer to unseen sensors remains unclear. In this section, we study this question for the first time to the best of our knowledge.

For unseen sensors, FTP-1 trains the sensor-specific tactile encoder from scratch, while reusing pre-trained components shared across sensors, including the tactile expert, the shared Transformer chunk for image-type tactile inputs [18], and functional-area embeddings. We first evaluate downstream

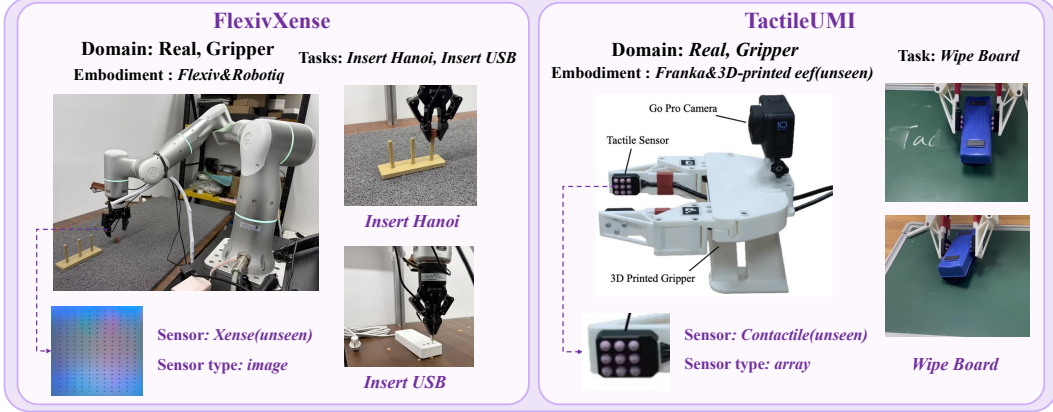


Figure 6: Overview of the finetuning experiment for unseen sensors during pretraining.

finetuning performance, then conduct ablations to verify whether the gains come from pretrained tactile knowledge.

4.1 Transfer to Unseen Tactile Sensor Setups

To evaluate FTP-1 on unseen tactile sensors, we distribute FTP-1 checkpoints to two new institutions with two new setups: FlexivXense with Xense image tactile sensors and TactileUMI [9] with Contactile array sensors [19], shown in Fig. 6. We also provide the pretraining sensors most similar to these two evaluation sensors in App. E.4. These setups include two fine-grained insertion tasks, Insert Hanoi (inserting a small circle hanoi toy to the pillar) and Insert USB for the FlexivXense domain (100 finetuning demonstrations per-task), and one press-control task, Wipe Board for the TactileUMI domain (50 demonstrations per-task). Hardware details are provided in App. E.1. Examples of policy rollouts are provided in App. E.5.

Evaluation Results. Tab. 3 reports the evaluation results on two unseen tactile sensor setups. Our method achieves the highest success rate among all baselines, reaching 46.6%. Compared with $\text{FTP-}\pi_{0.5}$ (15%), it improves the success rate by +31.6%, demonstrating that pretraining on large-scale heterogeneous tactile manipulation data effectively improves finetuning performance under unseen sensor settings. In contrast, $\pi_{0.5}$ performs comparably to $\text{FTP-}\pi_{0.5}$, suggesting that simply adding a tactile branch does not necessarily help without an appropriate modality-fusion prior and pretraining knowledge.

Table 3: Real-robot results on unseen tactile sensor setups with success rate as the metric.

Method	FlexivXense		TactileUMI	Avg.
	Insert Hanoi	Insert USB	Wipe Board	
$\pi_{0.5}$ [5]	25	0	20	15.0
Tactile-VLA [9]	0	10	15	8.3
$\text{FTP-}\pi_{0.5}$	5	10	30	15.0
FTP-1	55	30	55	46.6

For Insert Hanoi, FTP-1 and $\pi_{0.5}$ exhibit recovery behaviors, while the other baselines do not. This recovery ability likely benefits from large-scale pretraining. Moreover, FTP-1 shows reactive insertion control: when the circular Hanoi piece is misaligned with the pillar, FTP-1 slows down the insertion motion based on tactile feedback, whereas $\pi_{0.5}$ does not, often leading to failure. For Insert USB, the task is difficult to master with only 100 demonstrations, making it a good test of data efficiency. Compared with FTP-1, other models are less stable and sometimes exhibit small shaking motions during insertion, which reduces their success rates. For Wipe Board, we observe a similar trend to the previous Sharpa&Dexmate tasks: other models struggle to maintain stable pressing force and may lose tight contact with the board during wiping. Overall, these results show that FTP-1’s large-scale heterogeneous tactile pretraining provides a strong finetuning starting points, enabling better generalization, data efficiency, and reactive control in unseen tactile sensor settings.

4.2 Improvements Come from Pretrained Tactile Knowledge

The previous section shows that FTP-1 provides a strong initialization for tactile policy learning and improves downstream finetuning performance, even for unseen tactile sensors. However, it remains

unclear whether these gains truly come from pretrained tactile knowledge. We therefore examine two possible explanations:

- **Hypothesis 1 (Data Distribution):** The gains arise because the FTP-1 pretraining data is closer to the downstream task distribution, making finetuning easier.
- **Hypothesis 2 (Transferable Knowledge):** FTP-1 learns transferable tactile manipulation knowledge, providing better initialization for downstream tactile policy finetuning.

To verify this, we pretrain a No-Tactile-Pretraining checkpoint (NTP) using the same setup as FTP-1, including the same pretraining data and optimization settings, but excluding tactile inputs and tactile-related architecture. During finetuning, we add the same tactile-related architecture as FTP-1 during finetuning and refer to the resulting model as NTP-1. We evaluate NTP-1 on UniVTAC [27] (seen sensor) and FlexivXense (unseen sensor), using the same finetuning setup as previous section.

Results are shown in Fig. 7, with detailed results provided in App. E.6. All models use exactly the same architecture, differing only in their pre-trained checkpoints. On UniVTAC, NTP-1 outperforms FTP- $\pi_{0.5}$, suggesting that the distribution of the FTP-1-Dataset may be closer to UniVTAC. However, NTP-1 still underperforms FTP-1, indicating that FTP-1 pretraining provides a useful initialization for the tactile branch. On FlexivXense, FTP-1 substantially outperforms NTP-1 by +37.5%, demonstrating that tactile-based pretraining is essential for transferring to the FlexivXense domain. Without tactile-branch pretraining, NTP-1 shows much less robustness to tactile changes and produces unstable actions during key insertion stages. Together, these results support Hypothesis 2 (Transferable Knowledge): the tactile branch of FTP-1 benefits from pretraining, which encodes general tactile manipulation knowledge that can be transferred to downstream contact-rich tasks.

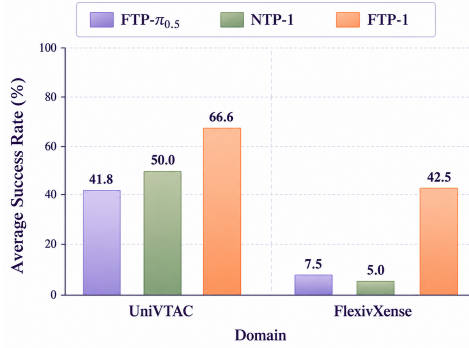


Figure 7: Comparison between FTP-1 and NTP-1 on UniVTAC and FlexivXense.

5 Related Work

A detailed review of related works is provided in App. A. Recent generalist robot policies [11, 1, 5, 2, 29] show strong transfer across tasks, embodiments, and environments, but largely omit tactile sensing. Existing tactile policy and Vision-Tactile-Language-Action (VTLA) methods [10, 9, 6, 30, 24] demonstrate the value of tactile feedback for contact-rich manipulation, but are typically tied to specific sensors, tasks, or embodiments. Tactile representation and policy pretraining methods [31, 32, 28, 18, 8, 33] improve tactile perception or downstream control, but do not establish an end-to-end sensor-agnostic generalist tactile policy across heterogeneous tactile hardware.

FTP-1 addresses this gap by extending the generalist policy paradigm to tactile manipulation. Rather than treating tactile feedback as a sensor-specific add-on for policies, FTP-1 studies whether tactile manipulation skills can be pretrained at scale and transferred across heterogeneous sensors and embodiments. This positions FTP-1 as a foundation-policy baseline for tactile manipulation, with evaluation covering both seen and unseen tactile-sensor setups.

6 Conclusion

We present FTP-1, the first generalist foundation tactile policy pretrained for diverse sensors and embodiments. By pretraining on a large-scale heterogeneous tactile manipulation dataset aggregated from 26 sources and covering 21 sensors, FTP-1 learns transferable tactile manipulation skills. Experiments on 5 hardware setups show that FTP-1 improves finetuning performance on both seen and unseen sensors. This provides a shared model-level starting point for tactile policy learning.

Limitations and Future Directions. As a preliminary exploration of generalist tactile policy learning, FTP-1 mainly focuses on general tactile perception and does not yet address tactile- or force-based servoing and control. Extending our heterogeneous encoding framework to future tactile prediction [8, 14] and prediction-based low-level control is a promising direction [34, 35, 36]. Another limitation is that the scale and diversity of our pretraining dataset remain limited. Building larger aggregated datasets [37] and further scaling up pretraining are left for future work.

Acknowledgments

This work is supported by Shanghai Qi Zhi Institute Innovation Program and the Tsinghua University Dushi Program. It is also largely supported by Sharpa Pte Ltd. through hardware and computation resources and the Sharpa North-FTP-1 datasets. The EgoTac egocentric human dataset and AetherGlove used in this study are provided by Aether Embodied Technology (Beijing) Co., Ltd. The OmniSharingDB human data is provided by PaXini Technology (Shenzhen) Co., Ltd.

References

- [1] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. pi0: A vision-language-action flow model for general robot control. [arXiv preprint arXiv:2410.24164](#), 2024.
- [2] J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. [arXiv preprint arXiv:2503.14734](#), 2025.
- [3] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. [arXiv preprint arXiv:2304.13705](#), 2023.
- [4] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. [The International Journal of Robotics Research](#), 44(10-11):1684–1704, 2025.
- [5] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, et al. pi0.5: a vision-language-action model with open-world generalization. [arXiv preprint arXiv:2504.16054](#), 2025.
- [6] Z. Cheng, Y. Zhang, W. Zhang, H. Li, K. Wang, L. Song, and H. Zhang. Omnivtla: Vision-tactile-language-action model with semantic-aligned tactile sensing. [arXiv preprint arXiv:2508.08706](#), 2025.
- [7] Q. Liu, Y. Cui, Z. Sun, G. Li, J. Chen, and Q. Ye. Vtdexmanip: A dataset and benchmark for visual-tactile pretraining and dexterous manipulation with reinforcement learning. In [The Thirteenth International Conference on Learning Representations](#), 2025.
- [8] Y. Zheng, S. Gu, W. Li, Y. Zheng, Y. Zang, S. Tian, X. Li, C. Hao, C. Gao, S. Liu, et al. Omnivta: Visuo-tactile world modeling for contact-rich robotic manipulation. [arXiv preprint arXiv:2603.19201](#), 2026.
- [9] J. Huang, S. Wang, F. Lin, Y. Hu, C. Wen, and Y. Gao. Tactile-vla: unlocking vision-language-action model’s physical knowledge for tactile generalization. [arXiv preprint arXiv:2507.09160](#), 2025.
- [10] L. Heng, H. Geng, K. Zhang, P. Abbeel, and J. Malik. Vitacformer: Learning cross-modal representation for visuo-tactile dexterous manipulation. [arXiv preprint arXiv:2506.15953](#), 2025.
- [11] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. [arXiv preprint arXiv:2406.09246](#), 2024.
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. [Advances in neural information processing systems](#), 30, 2017.
- [13] G. Zhang, Q. Xu, H. Zhang, J. Ma, L. He, Y. Bao, Z. Ping, Z. Yuan, C. Lu, C. Yuan, et al. Unidex: A robot foundation suite for universal dexterous hand control from egocentric human videos. [arXiv preprint arXiv:2603.22264](#), 2026.
- [14] Y. Niu, Z. Fang, B. Chen, S. Zhou, R. Senthilkumaran, H. Zhang, B. Chen, C. Qiu, H. E. Tseng, J. Francis, et al. Learning versatile humanoid manipulation with touch dreaming. [arXiv preprint arXiv:2604.13015](#), 2026.
- [15] L. Wang, X. Chen, J. Zhao, and K. He. Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers. [Advances in neural information processing systems](#), 37: 124420–124450, 2024.

- [16] W. Yuan, S. Dong, and E. H. Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762, 2017.
- [17] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.
- [18] J. Zhao, Y. Ma, L. Wang, and E. H. Adelson. Transferable tactile transformers for representation learning across diverse sensors and tasks. *arXiv preprint arXiv:2406.13640*, 2024.
- [19] E. Velasco-Sanchez, J. Castaño-Amoros, P. Gil, and F. Torres. Touch-based effector control to track 3d surfaces. In *2025 IEEE 30th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–6. IEEE, 2025.
- [20] J. Wu. Introduction to convolutional neural networks. *National Key Lab for Novel Software Technology. Nanjing University. China*, 5(23):495, 2017.
- [21] J. Huang, Y. Ye, Y. Gong, X. Zhu, Y. Gao, and K. Zhang. Spatially anchored tactile awareness for robust dexterous manipulation. *arXiv preprint arXiv:2510.14647*, 2025.
- [22] L. Beyer, A. Steiner, A. S. Pinto, A. Kolesnikov, X. Wang, D. Salz, M. Neumann, I. Alabdulmohsin, M. Tschannen, E. Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024.
- [23] R. Zhang, J. Han, C. Liu, P. Gao, A. Zhou, X. Hu, S. Yan, P. Lu, H. Li, and Y. Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023.
- [24] Y. Li, H. Jiang, J. Xia, H. Zhang, J. Du, Y. Zhou, J. Zeng, C. Hao, J. Ren, Q. Yu, et al. Forcevla2: Unleashing hybrid force-position control with force awareness for contact-rich manipulation. *arXiv preprint arXiv:2603.15169*, 2026.
- [25] H. Fang, S. Tang, M. Mei, H. Qin, Z. He, J. Chen, Y. Feng, C. Wang, W. Liu, Z. He, et al. Force policy: Learning hybrid force-position control policy under interaction frame for contact-rich manipulation. *arXiv preprint arXiv:2602.22088*, 2026.
- [26] T. Tang, X. Ji, W. Xing, C. Hao, W. Xu, L. Shao, C. Lu, Q. Yu, J. Pang, and K. Zhang. Towards human-like manipulation through rl-augmented teleoperation and mixture-of-dexterous-experts v1a. *arXiv preprint arXiv:2603.08122*, 2026.
- [27] B. Chen, W. Wan, T. Chen, X. Guo, C. Xu, Y. Qi, H. Zhang, L. Wu, T. Xu, Z. Li, et al. Univtac: A unified simulation platform for visuo-tactile manipulation data generation, learning, and benchmarking. *arXiv preprint arXiv:2602.10093*, 2026.
- [28] A. George, S. Gano, P. Katragadda, and A. B. Farimani. Vital pretraining: Visuo-tactile pretraining for tactile and non-tactile manipulation policies. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 258–264. IEEE, 2025.
- [29] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. In *International Conference on Learning Representations*, volume 2025, pages 29982–30009, 2025.
- [30] C. Zhang, P. Hao, X. Cao, X. Hao, S. Cui, and S. Wang. Vtla: Vision-tactile-language-action model with preference learning for insertion manipulation. *Biomimetic Intelligence and Robotics*, page 100333, 2026.
- [31] C. Higuera, A. Sharma, C. K. Bodduluri, T. Fan, P. Lancaster, M. Kalakrishnan, M. Kaess, B. Boots, M. Lambeta, T. Wu, et al. Sparsh: Self-supervised touch representations for vision-based tactile sensing. *arXiv preprint arXiv:2410.24090*, 2024.

- [32] R. Feng, J. Hu, W. Xia, T. Gao, A. Shen, Y. Sun, B. Fang, and D. Hu. Anytouch: Learning unified static-dynamic representation across multiple visuo-tactile sensors. [arXiv preprint arXiv:2502.12191](#), 2025.
- [33] L. Wu, C. Yu, J. Ren, L. Chen, Y. Jiang, R. Huang, G. Gu, and H. Li. Freetacman: Robot-free visuo-tactile data collection system for contact-rich manipulation. [arXiv preprint arXiv:2506.01941](#), 2025.
- [34] Z. Xu, Y. Wang, B. Abbatematteo, J. Preechayasomboon, S. Chan, N. Colonnese, and A. H. Memar. Contact-grounded policy: Dexterous visuotactile policy with generative contact grounding. [arXiv preprint arXiv:2603.05687](#), 2026.
- [35] D. Zhang, C. Yuan, C. Wen, H. Zhang, J. Zhao, and Y. Gao. Kinedex: Learning tactile-informed visuomotor policies via kinesthetic teaching for dexterous manipulation. [arXiv preprint arXiv:2505.01974](#), 2025.
- [36] P. Zhi, P. Li, J. Yin, B. Jia, and S. Huang. Learning a unified policy for position and force control in legged loco-manipulation. [arXiv preprint arXiv:2505.20829](#), 2025.
- [37] A. O’Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.
- [38] J. Barreiros, A. Beaulieu, A. Bhat, R. Cory, E. Cousineau, H. Dai, C.-H. Fang, K. Hashimoto, M. Z. Irshad, M. Itkina, et al. A careful examination of large behavior models for multitask dexterous manipulation. *Science Robotics*, 11(113):eaea6201, 2026.
- [39] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine. Fast: Efficient action tokenization for vision-language-action models. [arXiv preprint arXiv:2501.09747](#), 2025.
- [40] S. Liu, B. Li, K. Ma, L. Wu, H. Tan, X. Ouyang, H. Su, and J. Zhu. Rdt2: Exploring the scaling limit of umi data towards zero-shot cross-embodiment generalization. [arXiv preprint arXiv:2602.03310](#), 2026.
- [41] G. R. Team, S. Abeyruwan, J. Ainslie, J.-B. Alayrac, M. G. Arenas, T. Armstrong, A. Balakrishna, R. Baruch, M. Bauza, M. Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. [arXiv preprint arXiv:2503.20020](#), 2025.
- [42] G. R. Team, A. Abdolmaleki, S. Abeyruwan, J. Ainslie, J.-B. Alayrac, M. G. Arenas, A. Balakrishna, N. Batchelor, A. Bewley, J. Bingham, et al. Gemini robotics 1.5: Pushing the frontier of generalist robots with advanced embodied reasoning, thinking, and motion transfer. [arXiv preprint arXiv:2510.03342](#), 2025.
- [43] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. [arXiv preprint arXiv:2403.12945](#), 2024.
- [44] S. Wu, X. Liu, S. Xie, P. Wang, X. Li, B. Yang, Z. Li, K. Zhu, H. Wu, Y. Liu, et al. Robocoin: An open-sourced bimanual robotic data collection for integrated manipulation. [arXiv preprint arXiv:2511.17441](#), 2025.
- [45] R. Zheng, D. Niu, Y. Xie, J. Wang, M. Xu, Y. Jiang, F. Castañeda, F. Hu, Y. L. Tan, L. Fu, et al. Egoscale: Scaling dexterous manipulation with diverse egocentric human data. [arXiv preprint arXiv:2602.16710](#), 2026.
- [46] R. Yang, Q. Yu, Y. Wu, R. Yan, B. Li, A.-C. Cheng, X. Zou, Y. Fang, X. Cheng, R.-Z. Qiu, et al. Egovla: Learning vision-language-action models from egocentric human videos. [arXiv preprint arXiv:2507.12440](#), 2025.

- [47] C. Yuan, R. Zhou, M. Liu, Y. Hu, S. Wang, L. Yi, C. Wen, S. Zhang, and Y. Gao. Motiontrans: Human vr data enable motion-level learning for robotic manipulation policies. [arXiv preprint arXiv:2509.17759](#), 2025.
- [48] X. Cai, R.-Z. Qiu, G. Chen, L. Wei, I. Liu, T. Huang, X. Cheng, and X. Wang. In-n-on: Scaling egocentric manipulation with in-the-wild and on-task data. [arXiv preprint arXiv:2511.15704](#), 2025.
- [49] C. Yuan, C. Wen, T. Zhang, and Y. Gao. General flow as foundation affordance for scalable robot learning. [arXiv preprint arXiv:2401.11439](#), 2024.
- [50] Q. Li, Y. Deng, Y. Liang, L. Luo, L. Zhou, C. Yao, L. Zeng, Z. Feng, H. Liang, S. Xu, et al. Scalable vision-language-action model pretraining for robotic manipulation with real-life human activity videos. [arXiv preprint arXiv:2510.21571](#), 2025.
- [51] Y. Fu, N. Chen, J. Zhao, S. Shan, G. Yao, P. Wang, Z. Wang, and S. Zhang. Metis: Multi-source egocentric training for integrated dexterous vision-language-action model. [arXiv preprint arXiv:2511.17366](#), 2025.
- [52] J. Lyu, K. Liu, X. Zhang, H. Liao, Y. Feng, W. Zhu, T. Shen, J. Chen, J. Zhang, Y. Dong, et al. Lda-1b: Scaling latent dynamics action model via universal embodied data ingestion. [arXiv preprint arXiv:2602.12215](#), 2026.
- [53] D. Driess, J. Springenberg, B. Ichter, L. Yu, A. Li-Bell, K. Pertsch, A. Ren, H. Walke, Q. Vuong, L. X. Shi, et al. Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. [Advances in Neural Information Processing Systems](#), 38:102867–102888, 2026.
- [54] Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li. Univla: Learning to act anywhere with task-centric latent actions. [arXiv preprint arXiv:2505.06111](#), 2025.
- [55] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, et al. World action models are zero-shot policies. [arXiv preprint arXiv:2602.15922](#), 2026.
- [56] T. Yuan, Z. Dong, Y. Liu, and H. Zhao. Fast-wam: Do world action models need test-time future imagination? [arXiv preprint arXiv:2603.16666](#), 2026.
- [57] H. Bi, H. Tan, S. Xie, Z. Wang, S. Huang, H. Liu, R. Zhao, Y. Feng, C. Xiang, Y. Rong, et al. Motus: A unified latent action world model. [arXiv preprint arXiv:2512.13030](#), 2025.
- [58] Y. Hu, Y. Guo, P. Wang, X. Chen, Y.-J. Wang, J. Zhang, K. Sreenath, C. Lu, and J. Chen. Video prediction policy: A generalist robot policy with predictive visual representations. [arXiv preprint arXiv:2412.14803](#), 2024.
- [59] F. Lin, R. Nai, Y. Hu, J. You, J. Zhao, and Y. Gao. Onetwovla: A unified vision-language-action model with adaptive reasoning. [arXiv preprint arXiv:2505.11917](#), 2025.
- [60] F. Lin, K. Arora, J. Mercat, H. Nishimura, P. Shah, C. Xu, M. Zhang, M. Zolotas, M. Angeles, O. Pfannenstiehl, et al. A systematic study of data modalities and strategies for co-training large behavior models for robot manipulation. [arXiv preprint arXiv:2602.01067](#), 2026.
- [61] Z. Cheng, Y. Zhao, K. Wang, H. Zhang, and L. Song. Taco: A benchmark for lossless and lossy codecs of heterogeneous tactile data. In [The Fourteenth International Conference on Learning Representations](#), 2026.
- [62] Y. Wi, J. Yin, E. Xiang, A. Sharma, J. Malik, M. Mukadam, N. Fazeli, and T. Hellebrekers. Tac-align: Human-to-robot policy transfer via tactile alignment. [arXiv preprint arXiv:2602.13579](#), 2026.

- [63] F. Jia, X. Niu, S. Yang, Q. Ben, T. Huang, J. Wang, J. Pang, et al. Feel robot feels: Tactile feedback array glove for dexterous manipulation. [arXiv preprint arXiv:2603.28542](#), 2026.
- [64] B. Huang, Y. Wang, X. Yang, Y. Luo, and Y. Li. 3d-vitac: Learning fine-grained manipulation with visuo-tactile sensing. [arXiv preprint arXiv:2410.24091](#), 2024.
- [65] H. Xue, J. Ren, W. Chen, G. Zhang, Y. Fang, G. Gu, H. Xu, and C. Lu. Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation. [arXiv preprint arXiv:2503.02881](#), 2025.
- [66] K. Yu, Y. Han, Q. Wang, V. Saxena, D. Xu, and Y. Zhao. Mimictouch: Leveraging multi-modal human tactile demonstrations for contact-rich manipulation. [arXiv preprint arXiv:2310.16917](#), 2023.
- [67] Z. He, H. Fang, J. Chen, H.-S. Fang, and C. Lu. Foar: Force-aware reactive policy for contact-rich robotic manipulation. [IEEE Robotics and Automation Letters](#), 2025.
- [68] Y. Tian, S. Cheng, T. Wei, T. Zhou, Y. Zhang, Z. Liu, Q. Han, Z. Yuan, and H. Xu. Vitas: Visual tactile soft fusion contrastive learning for visuomotor learning. [arXiv preprint arXiv:2602.11643](#), 2026.
- [69] H. Choi, Y. Hou, C. Pan, S. Hong, A. Patel, X. Xu, M. R. Cutkosky, and S. Song. In-the-wild compliant manipulation with umi-ft. [arXiv preprint arXiv:2601.09988](#), 2026.
- [70] F. Liu, C. Li, Y. Qin, J. Xu, P. Abbeel, and R. Chen. Vitamin: Learning contact-rich tasks through robot-free visuo-tactile manipulation interface. [arXiv preprint arXiv:2504.06156](#), 2025.
- [71] C. Li, C. Liu, D. Wang, S. Zhang, L. Li, Z. Zeng, F. Liu, J. Xu, and R. Chen. Vitamin-b: A reliable and efficient visuo-tactile bimanual manipulation interface. [arXiv preprint arXiv:2511.05858](#), 2025.
- [72] Y. Xu, L. Wei, P. An, Q. Zhang, and Y.-L. Li. exumi: Extensible robot teaching system with action-aware task-agnostic tactile representation. [arXiv preprint arXiv:2509.14688](#), 2025.
- [73] C. Higuera, A. Sharma, T. Fan, C. K. Bodduluri, B. Boots, M. Kaess, M. Lambeta, T. Wu, Z. Liu, F. R. Hogan, et al. Tactile beyond pixels: Multisensory touch representations for robot manipulation. In [Conference on Robot Learning](#), pages 105–123. PMLR, 2025.
- [74] J. Bi, K. Y. Ma, C. Hao, M. Z. Shou, and H. Soh. Vla-touch: Enhancing vision-language-action models with dual-level tactile feedback. [arXiv preprint arXiv:2507.17294](#), 2025.
- [75] K. Zhang, H. Zhang, Z. Xu, Z. Zhang, M. R. I. Prince, X. Li, X. Han, Y. Zhou, A. Ajoudani, and Y. She. Tacvla: Contact-aware tactile fusion for robust vision-language-action manipulation. [arXiv preprint arXiv:2603.12665](#), 2026.
- [76] J. Yu, H. Liu, Q. Yu, J. Ren, C. Hao, H. Ding, G. Huang, G. Huang, Y. Song, P. Cai, et al. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation. [Advances in Neural Information Processing Systems](#), 38:93409–93439, 2026.
- [77] W. Wu, F. Lu, Y. Wang, S. Yang, S. Liu, F. Wang, Q. Zhu, H. Sun, Y. Wang, S. Ma, et al. A pragmatic vla foundation model. [arXiv preprint arXiv:2601.18692](#), 2026.
- [78] G. A. Team. Gen-0: Embodied foundation models that scale with physical interaction. [Generalist AI Blog](#), 2025. <https://generalistai.com/blog/nov-04-2025-GEN-0>.
- [79] Q. Liu, Q. Ye, Z. Sun, Y. Cui, G. Li, and J. Chen. Masked visual-tactile pre-training for robot manipulation. In [2024 IEEE International Conference on Robotics and Automation \(ICRA\)](#), pages 13859–13875. IEEE, 2024.

- [80] N. Cheng, J. Xu, C. Guan, J. Gao, W. Wang, Y. Li, F. Meng, J. Zhou, B. Fang, and W. Han. Touch100k: A large-scale touch-language-vision dataset for touch-centric multimodal representation. *Information Fusion*, 124:103305, 2025.
- [81] X. Zhu, B. Huang, and Y. Li. Touch in the wild: Learning fine-grained manipulation with a portable visuo-tactile gripper. *Advances in Neural Information Processing Systems*, 38: 153783–153812, 2026.
- [82] Z. Sun, Z. Shi, J. Chen, Q. Liu, Y. Cui, J. Chen, and Q. Ye. Vtao-bimanip: Masked visual-tactile-action pre-training with object understanding for bimanual dexterous manipulation. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3201–3208. IEEE, 2025.
- [83] L. Wu, J. Ren, C. Jiang, J. Zhou, S. Peng, R. Huang, G. Gu, L. Chen, and H. Li. Tamen: Tactile-aware manipulation engine for closed-loop data collection in contact-rich tasks. *arXiv preprint arXiv:2604.07335*, 2026.
- [84] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. *arXiv preprint arXiv:2402.10329*, 2024.
- [85] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
- [86] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.
- [87] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [88] J. Liu, J. Su, X. Yao, Z. Jiang, G. Lai, Y. Du, Y. Qin, W. Xu, E. Lu, J. Yan, et al. Muon is scalable for llm training. *arXiv preprint arXiv:2502.16982*, 2025.
- [89] Y. R. Song, J. Li, R. Fu, D. Murphy, K. Zhou, R. Shiv, Y. Li, H. Xiong, C. E. Owens, Y. Du, et al. Opentouch: Bringing full-hand touch to real-world interaction. *arXiv preprint arXiv:2512.16842*, 2025.
- [90] Z. Zhao, H. Jing, X. Liu, J. Mao, A. Jha, H. Yang, R. Xue, S. Zakharov, V. Guizilini, and Y. Wang. Humanoid everyday: A comprehensive robotic dataset for open-world humanoid manipulation. *arXiv preprint arXiv:2510.08807*, 2025.
- [91] M. Xu, H. Zhang, Y. Hou, Z. Xu, L. Fan, M. Veloso, and S. Song. Dexumi: Using human hand as the universal manipulation interface for dexterous manipulation. *arXiv preprint arXiv:2505.21864*, 2025.
- [92] H.-S. Fang, H. Fang, Z. Tang, J. Liu, C. Wang, J. Wang, H. Zhu, and C. Lu. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. *arXiv preprint arXiv:2307.00595*, 2023.
- [93] D. Sliwowski, S. Jadav, S. Stanovcic, J. Orbik, J. Heidersberger, and D. Lee. Reassemble: A multimodal dataset for contact-rich robotic assembly and disassembly. *arXiv preprint arXiv:2502.05086*, 2025.
- [94] Z. Xu, R. Uppuluri, X. Zhang, C. Fitch, P. G. Crandall, W. Shou, D. Wang, and Y. She. Unit: Data efficient tactile representation with generalization to unseen objects. *IEEE Robotics and Automation Letters*, 2025.
- [95] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Appendix

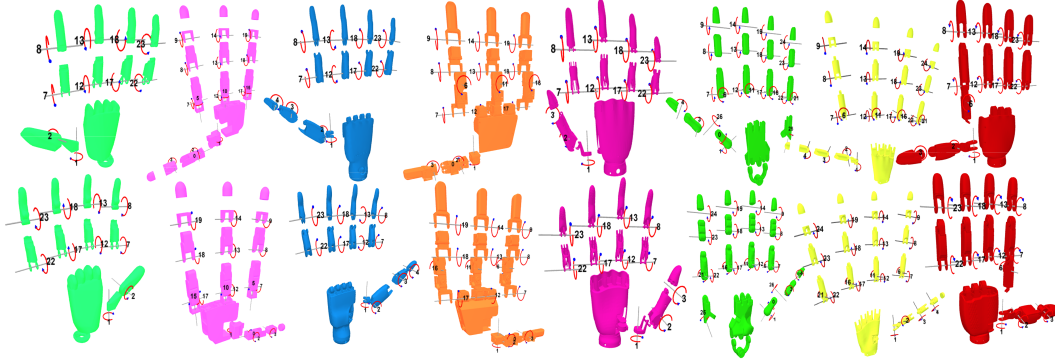


Figure 8: Joint mappings of different robotic hands used in FAAS. From left to right are Ability, Allegro, Inspire, Leap, Oymotion, Shadow, Wuji, and Xhand. The two rows show different views of the joint mappings on the right hand. Refer and copied from the original UniDex [13] paper.

A Detailed Related Works

A.1 Generalist Policy Learning for Robotic Manipulation

Recent progress in robot learning has shifted from task-specific policies [4, 3] toward the generalist policy paradigm. Generalist policies [11, 37, 38, 5, 2, 39, 29, 40, 15, 41, 42] are trained on large-scale robotic datasets [37, 43, 44] and can rapidly adapt to new environments, embodiments, and tasks. Beyond robot demonstrations, heterogeneous learning signals such as human motion [45, 46, 47, 48, 49, 50, 51, 13], semantic latent tokens [52, 53, 54], video generation [55, 56, 57, 58], and vision-language answering [5, 59, 60] have also been incorporated into training frameworks to improve generalization. However, existing generalist policies largely omit tactile inputs, which are crucial for contact-rich manipulation.

A.2 Tactile Policy Learning for Contact-Rich Manipulation

The absence of generalist tactile policies is largely due to the heterogeneity of tactile hardware [61]. Across platforms, tactile observations vary in modality, making cross-sensor generalization challenging [31, 62, 63]. Nevertheless, recent sensor-specific visuo-tactile policy learning works [10, 64, 14, 65, 21, 66, 25, 67, 68, 69, 70, 71, 72, 73] show that tactile signals substantially improve contact-rich manipulation. In parallel, Vision-Tactile-Language-Action (VTLA) models [9, 74, 30, 75, 76, 24] augment pretrained vision-based foundation policies with tactile inputs [1, 77]. In contrast, FTP-1 is a natively tactile foundation model [78], rather than a task-specific policy or a vision-policy extension.

A.3 Large-Scale Tactile Representation Pretraining

Recent works have explored tactile representation pretraining to improve tactile perception for downstream tasks. One line of work [7, 79, 80, 81, 27, 18], including Sparsh [31], AnyTouch [32], OmniVTLa [8], and ViTaL [28], learns tactile encoders from vision-tactile pairs for downstream policy finetuning. However, these representations are not optimized end-to-end for robotic manipulation. Another line directly pretrains tactile policies on action-labeled tactile manipulation data, including VTAO-BiManip [82], FreeTacMan [33], OmniVTA [8], and TAMEn [83], but remains limited by sensor-specific setups and dataset diversity and scale. FTP-1 instead offers a sensor-agnostic generalist foundation policy with large-scale pretraining for diverse tactile robot setups.

B Details of FTP-1 Architecture

B.1 Unified Action Space (UAS)

Different embodiments use different action spaces for robotic control. For example, one robot may control a single arm in joint space, while another may require bimanual end-effector pose control. To handle this heterogeneity, FTP-1 adopts a Unified Action Space (UAS) that represents all robot commands as fixed-length sparse vectors [29]. For one control step, the predicted action is $\mathbf{a} = [\mathbf{a}^L, \mathbf{a}^R, \mathbf{a}^{ego}, \mathbf{a}^{sup}] \in \mathbb{R}^D$ where $\mathbf{a}^L$ and $\mathbf{a}^R$ denote the left- and right-arm control signals, $\mathbf{a}^{ego}$ denotes the head-pose control signal, and $\mathbf{a}^{sup}$ provides supplementary slots for additional controls such as locomotion or waist motion.

For each arm $b \in \{L, R\}$, the arm action is defined as $\mathbf{a}^b = [\mathbf{t}_w^b, \mathbf{r}_w^b, \mathbf{q}_{arm}^b, \mathbf{q}_{hand}^b]$, where $\mathbf{t}_w^b \in \mathbb{R}^3$ is the wrist translation, $\mathbf{r}_w^b \in \mathbb{R}^6$ is the 6D wrist rotation representation [84], $\mathbf{q}_{arm}^b \in \mathbb{R}^7$ denotes arm joints, and $\mathbf{q}_{hand}^b \in \mathbb{R}^{32}$ denotes canonical hand joint control slots. For $\mathbf{q}_{hand}^b$, following UniDex [13], we use a Function–Actuator–Aligned Space (FAAS) to unify the control action spaces of different end-effectors.

Brief Review of FAAS [13]. For the hand joint action space q_{hand} , we follow UniDex [13] and adopt the Function–Actuator–Aligned Space (FAAS), which unifies the action spaces of different end-effectors by assigning functionally similar joints to the same action slot. Fig. 8 shows the FAAS definitions for commonly used dexterous hands, reproduced from the original paper. By providing a universal action interface, FAAS enables new end-effectors to be integrated by simply defining their mappings to the predefined functional slots, thereby sharing the same action semantics and knowledge as those used during pretraining.

We further define the FAAS for the human hand in Fig. 9. For parallel grippers, we use an independent slot 28. For head control, $\mathbf{a}^{ego} = [\mathbf{t}^{ego}, \mathbf{r}^{ego}]$, where $\mathbf{t}^{ego} \in \mathbb{R}^3$ and $\mathbf{r}^{ego} \in \mathbb{R}^6$ denote the translation and rotation of the head pose.

For each embodiment, only the supported control slots are filled in the UAS. During training, we apply a mask $\mathbf{M} \in \{0, 1\}^D$ to exclude missing or unsupported action dimensions from the loss. With the UAS, FTP-1 always predicts actions in the same layout, enabling unified training across embodiments with different action spaces.

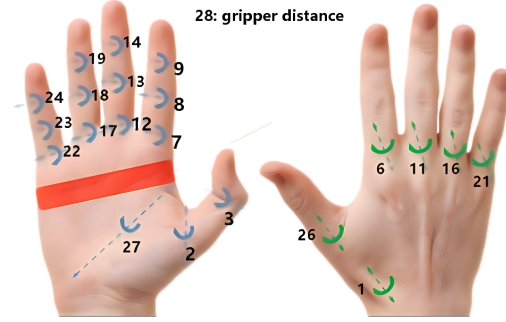


Figure 9: FAAS definition for the human hand.

B.2 Details of Heterogeneous Tactile Encoders

Here we provide the detailed architecture of tactile encoders for each type of tactile input.

Image-type inputs. For inputs of image shape (H, W, C) , we follow T3-Encoder [18]: each tactile image is resized to 224×224 , encoded by a sensor-specific Transformer encoder (depth=3, width=768, head=12), and then passed to a shared Transformer module (depth=9, width=768, head=12) for unified tokenization. The shared module is initialized from pretrained T3 weights [15], and the final [CLS] token is used as the tactile representation.

Array-type inputs. For inputs of shape (H, W, D) , where H, W is resolution of the array and D is the dimension of signal for each unit, we apply Fourier encoding [21] on the signal dimension, concatenate it with the original input, and process the result with a 3-layer CNN [20] followed by a 2-layer ReLU MLP to obtain the tactile token.

State-type inputs. For inputs of shape $(D,)$, we apply Fourier encoding [21], concatenate it with the original state, and use a 3-layer ReLU MLP to obtain the tactile token.

All tactile tokens are then normalized by LayerNorm [85], augmented with shared function-area embeddings, and projected by a 2-layer GELU MLP to match the tactile expert input dimension.

B.3 Adaptive RMSNorm Proprioception Injector

In preliminary experiments, we found that injecting proprioception through adaptive RMSNorm [5] improves generalization and robustness compared to using an independent proprioceptive token in the Transformer expert. We Fourier-encode [21] the proprioceptive state and process it with a 3-layer ReLU MLP. The resulting features are LayerNorm-normalized [85], concatenated with layer-normalized flow-matching timestep features from $\pi_{0.5}$ [5], and injected into the attention blocks via adaptive RMSNorm [5, 86].

B.4 Other Details

For state and action normalization, we adopt z-score normalization [1], which yielded better action generation quality than quantile-based normalization [5] in our experiments, especially for fine-grained action in contact-rich manipulation. The tactile expert is a Transformer with width 1024, depth 18, MLP dimension 4096, 8 attention heads, and head dimension 256. We train the model with AdamW [87]. We also try Muon [88] optimizer in our earlier experiments. Although we find that Muon optimizer improves convergence speed and offline action MSE significantly, it reduces generalization and robustness in real-robot rollouts, so we use AdamW in the final model.

C Details of FTP-1 Pretraining Datasets

In this section, we describe the FTP-1 pretraining datasets. An overview is provided in Tab. 4. The dataset is aggregated from 26 data sources, covering human demonstrations, dexterous-hand data, parallel-gripper data, and UMI-style data [84]. These sources include large-scale robotic manipulation datasets [92, 90, 33, 47, 81], human manipulation datasets with tactile annotations [89] and smaller-scale experimental datasets [91, 72, 70, 93, 94, 65, 74]. In addition, we collect a new robotic dataset, Sharpa North-FTP-1, using the Sharpa North hardware platform [10], containing approximately 4,000 long-horizon dexterous manipulation demonstrations.

FTP-1-Dataset spans 21 tactile sensors, including 7 image-type sensors (Sharpa DTC, ViTai-GF, FreeTacMan, exUMI, ViTaMin, GelSight-Mini, MCTac), 5 array-type sensors (AetherGlove, Open-TouchGlove, LinkerHandL6Touch, 3DViTac, uSkin), and 9 state-type sensors (PaxiniGlove, Uni-treeDex3, InspireHand, DexumiInspire, DexumiXHand, ATI Axia80-M20, FrankaTorque, AIDIN-FT, FlexivGripperTorque). For data sources with wrist-pose or head-pose annotations, we transform the poses into a unified coordinate direction definition. For language annotations, we use GPT-4o [95] to rewrite the original task instructions to increase linguistic diversity.

Since data sources vary substantially in scale, we adjust the sampling ratio of each source according to the scale in Tab. 4. After resampling, the mixture contains approximately 20% human-hand data, 30% dexterous-hand data, and 50% gripper data. In total, the dataset comprises around 3,000 hours of tactile-based manipulation data, covering diverse contact conditions, tasks, scenes, and embodiments. During pretraining, normalization statistics are computed independently for each dataset. Our heterogeneous tactile encoders are organized by tactile sensor rather than data source; therefore, data from the same sensor share the same tactile encoder across different data sources. Pretraining on this large-scale heterogeneous dataset enables FTP-1 to learn transferable tactile manipulation knowledge that adapts to diverse tactile sensors and embodiments.

Table 4: Overview of the FTP-1 pretraining datasets and sampling ratios. Data sources are grouped into three domains according to their end-effector types: human hand, dexterous hand, and gripper. To address data imbalance across sources, we resample data from each source according to the sampling scale shown in the table.

Data Source	Embodiment	Tactile / Force Sensor(s) (type)	Sample-Scale	Final-Percentage
Human Hand Domains (20%)				
EgoTac-Open	Human	AetherGlove (array)	0.89	8.2%
EgoTac-Kitchen	Human	AetherGlove (array)	1.63	5.5%
EgoTac-Hotel	Human	AetherGlove (array)	2.82	3.8%
OpenTouch [89]	Human	OpenTouchGlove (array)	11.09	1.5%
OmniSharingDB	Human	PaxiniGlove (state)	21.65	1.0%
Dexterous Hand Domains (30%)				
Sharpa North-FTP-1	DexHand	Sharpa DTC (image)	3.29	9.0%
HumanoidEveryday [90]	DexHand	UnitreeDex3 (state)	13.97	6.0%
OpenLETDex	DexHand	LinkerHandL6Touch (array)	14.14	6.0%
MotionTrans [47]	DexHand	InspireHand (state)	26.39	3.6%
DexumiInspire [91]	UMI+DexHand	DexumiInspire (state)	35.62	3.0%
DexumiXHand [91]	UMI+DexHand	DexumiXHand (state)	56.02	2.4%
Gripper / UMI Domains (50%)				
VTouch-D-WHEEL	Gripper	ViTai-GF (image)	2.77	5.7%
FreeTacMan [33]	UMI	FreeTacMan (image)	7.75	4.4%
TouchInTheWild [81]	UMI	3DViTac (array)	10.79	4.0%
VTouch-QINGLONG	Gripper	ViTai-GF (image)	12.73	3.9%
exUMI [72]	UMI	exUMI (image)	13.73	3.8%
RH20T-Cfg6 [92]	Gripper	ATIAxia80M20 (state)	13.88	3.8%
ViTaMIn [70]	UMI	ViTaMIn (image)	22.19	3.4%
RH20T-Cfg7 [92]	Gripper	uSkin (array); ATIAxia80M20 (state)	22.89	3.4%
RH20T-Cfg5 [92]	Gripper	FrankaTorque (state)	24.42	3.3%
REASSEMBLE [93]	Gripper	AIDIN-FT (state)	27.77	3.2%
Unit-Bimanual [94]	Gripper	GelSight-Mini (image)	63.93	2.6%
RDP [65]	Gripper	GelSight-Mini (image); MCTac (image); FlexivGripperTorque (state)	89.76	2.4%
Unit [94]	Gripper	GelSight-Mini (image)	132.83	2.2%
VLA-Touch [74]	Gripper	GelSight-Mini (image)	170.36	2.0%
RDP-Bimanual [65]	Gripper	GelSight-Mini (image); MCTac (image); FlexivGripperTorque (state)	199.20	1.9%

D Training Details

Pretraining Setting. FTP-1 is built on the $\pi_{0.5}$ codebase [5]. We initialize the vision encoder, tokenizer, vision-language expert, and action expert from $\pi_{0.5}$, and train the tactile encoder, tactile expert, adaptive RMSNorm proprioception injector, and action projector from scratch. Pretraining is performed on 48 NVIDIA H20 GPUs for 50k steps with global batch size 768 and a learning rate decayed from 1×10^{-4} to 5×10^{-5} . Performance saturates beyond 50k steps, likely due to limited tactile data diversity and the trade-off between preserving $\pi_{0.5}$ knowledge and learning tactile skills. Further increasing dataset diversity or co-training with $\pi_{0.5}$ datasets may lead to improved scaling behavior and we leave it to future works.

Pretraining Infrastructure. To support large-scale heterogeneous pretraining, we develop a new training infrastructure that automatically assigns data from different domains to separate GPUs. This ensures that samples within each GPU batch share the same data format, enabling efficient parallel training for large-scale heterogeneous datasets. Gradients of domain-specific modules are updated independently, while gradients of shared modules are merged before the joint update.

Finetuning Setting. Downstream finetuning uses 8 NVIDIA A800 GPUs for 20k steps per dataset, with batch size 64 and learning rate decayed from 5×10^{-5} to 5×10^{-6} . Other settings are the same as in pretraining.

Table 5: Overview of the evaluation setups. Components marked as **unseen** are not covered by FTP-1 large-scale heterogeneous pretraining.

Setup Name	Domain	Robot	End-effector	Tactile Sensor	Sensor Type
UniVTAC	Sim [27] (unseen), Gripper	Franka	Franka	GelSight-Mini [16]	image
Sharpa North	Real, Dexterous Hand	Sharpa North	Sharpa Wave	Sharpa DTC [10]	image
Sharpa&Dexmate	Real, Dexterous Hand	Dexmate (unseen)	Sharpa Wave	Sharpa DTC [10]	image
FlexivXense	Real, Gripper	Flexiv	Robotiq	Xense (unseen)	image
TactileUMI	Real, Gripper	Franka	3D-printed [9] (unseen)	Contactile [19] (unseen)	array

E Details of Evaluation

E.1 Hardware Setups

The detailed hardware setups are provided in Tab 5.

E.2 Evaluation Setups of Seen Sensors

We evaluate FTP-1 on three embodiments: UniVTAC [27] (simulation), and Sharpa North and Sharpa&Dexmate. These setups cover 2 tactile sensors, GelSight-Mini [16] and Sharpa DTC [10]; hardware details are summarized in App. E.1. Since these sensors are included in pretraining, both the tactile tokenizer and tactile expert are initialized from FTP-1 pretrained checkpoints.

For the UniVTAC simulation benchmark [27], we follow the original protocol and finetune each task with 50 official released demonstrations. We exclude two tasks that are unsuitable for evaluation. We exclude “Grasp Classify”, as it has already been solved perfectly by previous baselines. We also exclude “Insert HDMI” task because the motion-planning-based data collection in the original benchmark leads to perfect insertion in every demonstration, providing insufficient contact-feedback for evaluating tactile policy learning. The remaining 6 tasks cover two types of contact-rich behavior: in-hand manipulation (Lift Bottle, Lift Can, Put Bottle) and contact-aware insertion/extraction (Pull-out Key, Insert Hole, Insert Tube).

For Sharpa North, we finetune FTP-1 on 250-400 demonstrations across three long-horizon tasks: drawing a smiley face on a balloon (Draw Balloon), attaching a finger to a small damaged model hand (Fix Hand), and twisting a bottle cap with two hands (Twist Cap). For Fix Hand, we evaluate the task in two stages: first tearing off the small sticker from the damaged hand (Tear), and then attaching the finger to complete the repair (Finish). These tasks are deliberately challenging and are designed to probe the capability boundary of FTP-1, requiring deformable-object interaction, fine-grained contact control, small-object manipulation, bimanual coordination, and other complex contact-rich skills.

For Sharpa&Dexmate, we include two additional tasks: flipping a book (Flip Book) and wiping a dish (Wipe Dish). These tasks complement the evaluation suite by testing contact-rich pressing and force-control behaviors. Each task is finetuned with 100 demonstrations.

E.3 Visualization of Policy Rollouts for Seen-Sensor Evaluation

We provide representative rollout visualizations for the seen-sensor evaluation setups in Fig. 11, Fig. 12, and Fig. 13. Each row corresponds to one task and shows five key frames sampled from a successful rollout.

E.4 Details of Similar Sensors for Unseen Sensor Evaluation

In our unseen sensor evaluation setup, Figure 10 shows the pretraining sensors that are most similar to the downstream evaluation sensors. For Xense (image-type), the closest counterpart is GelSight-Mini [16]. For Contactile [19] (array-type), the closest counterpart is AetherGlove.

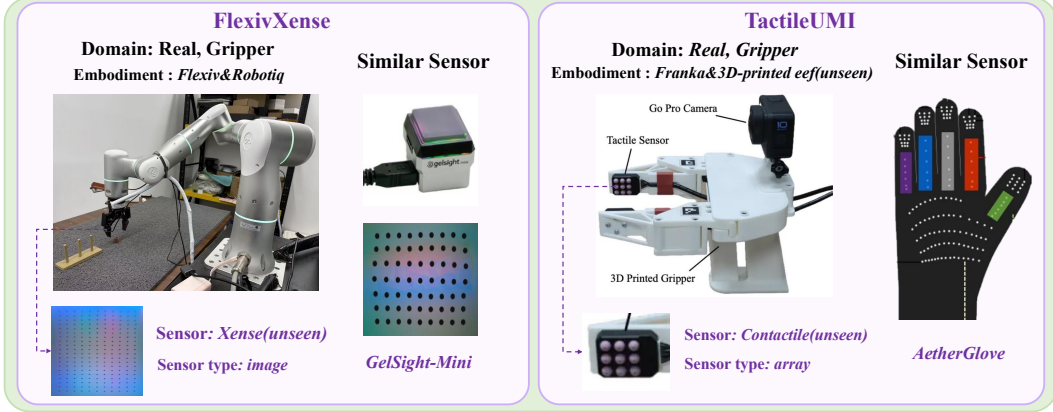


Figure 10: Similar sensors between the pretraining dataset and the downstream evaluation setups.



Figure 11: Representative rollouts on the UniVTAC simulation benchmark. Each row corresponds to one task and shows five key frames from a rollout.

E.5 Visualization of Policy Rollouts for Unseen-Sensor Evaluation

We provide representative rollout visualizations for the unseen-sensor evaluation setups in Fig. 14 and Fig. 15.

E.6 Details of Pretrained Tactile Knowledge Ablation

Detailed ablation results for pretrained tactile knowledge are provided in Tab. 6.

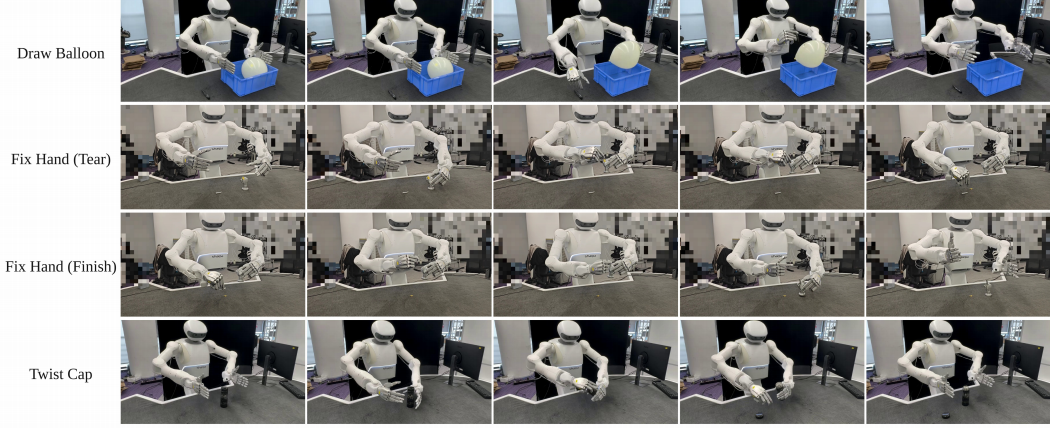


Figure 12: Representative rollouts on the Sharpa North setup, including Draw Balloon, Fix Hand (Tear), Fix Hand (Finish), and Twist Cap.

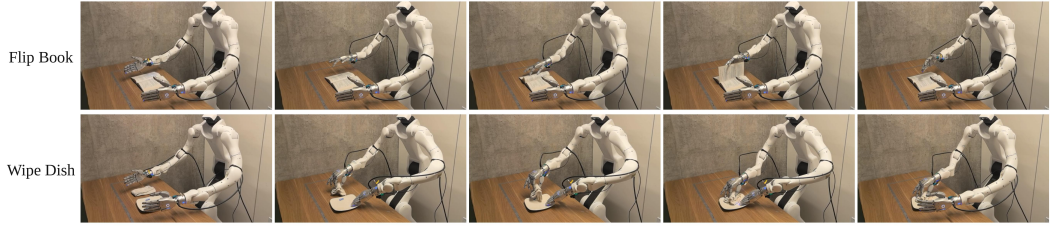


Figure 13: Representative rollouts on the Sharpa&Dexmate setup, including Flip Book and Wipe Dish.

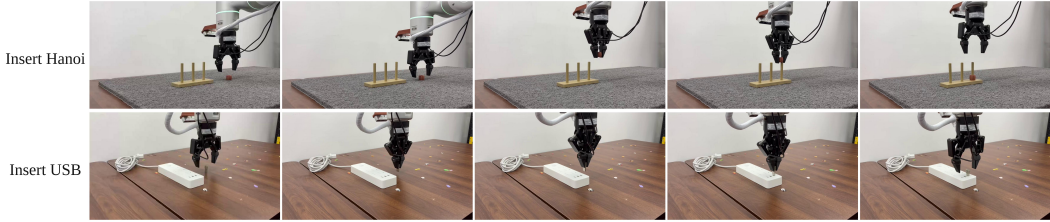


Figure 14: Representative rollouts on the FlexivXense setup, including Insert Hanoi and Insert USB.

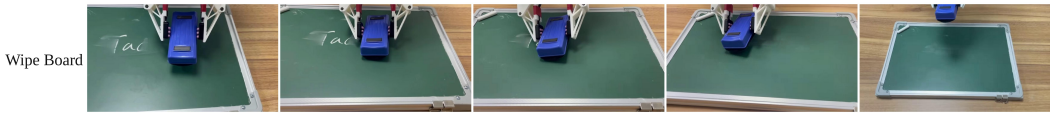


Figure 15: Representative rollouts on the TactileUMI setup for the Wipe Board task.

Table 6: Results of Pretrained Tactile Knowledge Ablation on the UniVTAC simulation benchmark. We report success rates (%). “Avg. w/o Lifts” excludes Lift Bottle and Lift Can. The best and second-best results are highlighted in **bold** and underline, respectively.

Method	Lift Bottle	Pull-out Key	Lift Can	Put Bottle	Insert Hole	Insert Tube	Avg.	Avg. w/o Lift
FTP-$\pi_{0.5}$	77	30	26	19	<u>47</u>	<u>72</u>	45.16	<u>42</u>
NTP-1	<u>88</u>	<u>38</u>	66	<u>32</u>	31	45	<u>50.00</u>	36.5
FTP-1	97	48	<u>65</u>	47	64	79	66.66	59.5