FACT: A Multimodal Force–Action–Track Foundation Model for Human-to-Robot Dexterity

Zhi Cao[1,2*], Yining Hong[2*], Jiaolong Yang[3], Yejin Choi[2], Li Fei-Fei[2], Jiajun Wu[2†], Huang Huang[2†]
[1]University of Michigan [2]Stanford University [3]Microsoft Research
[*]Equal contribution [†]Equal advising

Website: https://fact-1-0.github.io/

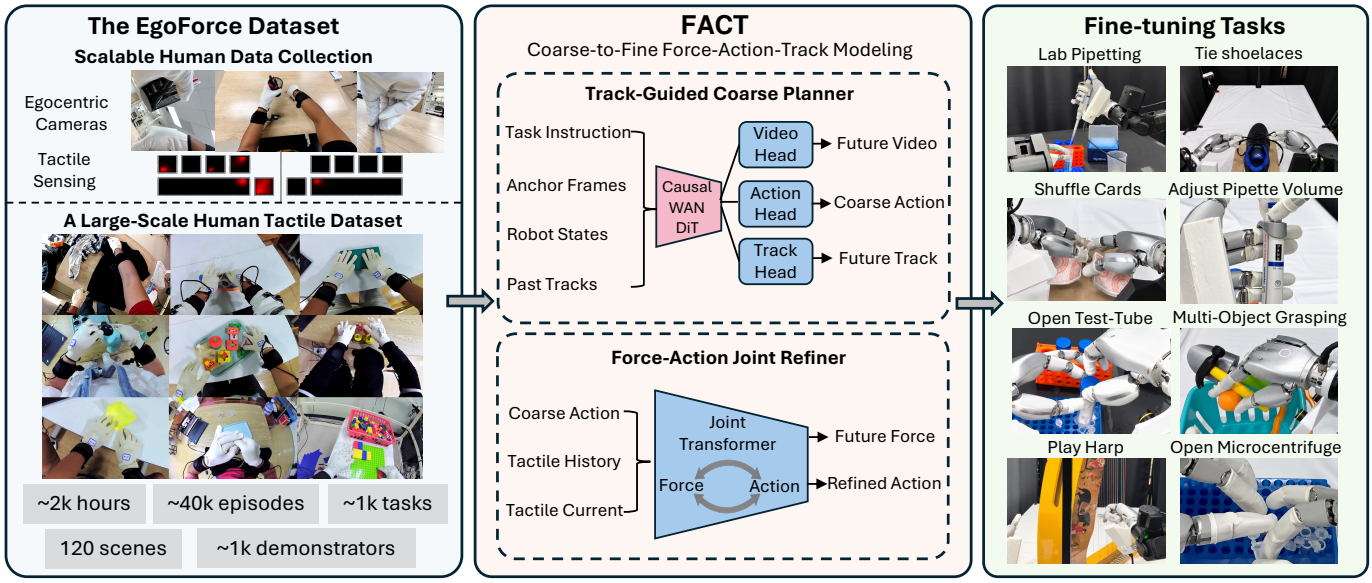


Fig. 1: FACT combines large-scale force-rich human experience with robot demonstrations for pretraining. Its coarse-to-fine architecture uses visual tracks and video prediction to shape coarse action, followed by joint force–action refinement for fine-grained execution. The pretrained model adapts to diverse tasks, from tool use, musical instruments, to precision laboratory procedures.

Abstract*—Human demonstrations offer a scalable alternative to costly robot data collection for dexterous manipulation, but most human datasets lack force measurements. Dexterous manipulation requires not only visual and proprioceptive understanding but also the ability to anticipate and regulate interaction forces. To scale force-aware human–robot co-training, we introduce *EgoForce*, a large-scale human tactile dataset capturing force-rich manipulation experience, and combine it with robot demonstrations for pretraining. We present FACT, a robot foundation model that cascades from prospective visual planning to force-aware refinement for dexterous manipulation. FACT comprises two jointly trained components. A Track-Guided Coarse Planner jointly predicts actions, future video, and visual tracks. Future video and visual tracks also provide a shared task representation that facilitates cross-embodiment knowledge transfer and coarse action generation. Building on this coarse plan, a Force–Action Joint Refiner jointly predicts future force and refines the coarse action, allowing anticipated force and action to shape each other during refinement. We evaluate the model on ten tasks requiring fine-grained coordination and force feedback, including tool use, musical instruments, and precision laboratory procedures. Using as few as 10 robot trajectories with 90 human trajectories for fine-tuning, FACT outperforms both baselines, with success-rate gains of up to 60 percentage points over T-Rex. Ablations show that force supervision in human pretraining data is crucial for downstream robot performance after fine-tuning, while visual track prediction and training with future-force prediction improve dexterous execution. Experiments varying the human–robot data ratio during fine-tuning further suggest that FACT learns from human demonstrations nearly as effectively as from robot demonstrations.

I. Introduction

Collecting diverse robot demonstrations for dexterous manipulation at scale is expensive, making human demonstrations an attractive source of manipulation experience. Most human manipulation datasets capture vision and motion without measuring contact forces. However, contact-rich dexterous manipulation often requires both prospective visual planning and force-aware refinement [1, 2]: where the hands go is guided by vision and task context, while how the fingers finish the motion is shaped by contact forces. The former determines the coarse structure of the motion; the latter governs its fine-grained execution. Force measurements can provide supervision for how

hand motion and physical interaction evolve together, allowing human experience to better support force-aware robot learning.

To support force-aware learning at scale, we develop a collection platform combining three egocentric views, bimanual tactile gloves, and automated force calibration to capture synchronized vision, hand motion, and tactile force during natural manipulation. Using this platform, we collect EgoForce, a large-scale human tactile dataset comprising approximately 2k hours of manipulation across 40k episodes, 1k tasks, and 1k demonstrators.

To transfer this force-rich human experience to robots, we introduce FACT, a dexterous foundation model that cascades from prospective visual motion planning to force-aware refinement for human-to-robot dexterity (Figure 1). The model comprises two jointly trained components for coarse motion planning and fine-grained force-aware execution. The Track-Guided Coarse Planner jointly predicts future video, visual tracks, and an action chunk. Building on the coarse plan, a Force–Action Joint Refiner takes the partially denoised action together with observed tactile signals and co-denoises the remaining action and future fingertip force through bidirectional self-attention, allowing action and force to reciprocally condition one another for action refinement. Future video and visual track prediction not only ground coarse action generation in anticipated visual motion but also provide an embodiment-agnostic task representation that encourages knowledge transfer between humans and robots.

We pretrain FACT on our large-scale EgoForce dataset together with a small set of robot demonstrations. We then fine-tune and evaluate it across 10 real-world tasks, spanning everyday skills such as tying shoelaces, precision laboratory procedures such as pipetting and microcentrifuge-tube manipulation, as well as tool use and musical instruments. Our ablations show that force supervision in human pretraining data provides the largest gains on tasks demanding precise contact and force control, while visual track prediction and training with future-force prediction improve dexterous execution. Experiments with different human–robot data ratios during fine-tuning suggest that human demonstrations can contribute nearly as much as robot demonstrations to policy learning, supporting human experience as a scalable source of dexterous manipulation skills.

Our contributions are threefold:

- We introduce FACT, a dexterous foundation model that cascades from track-based visual motion planning to force-aware refinement for dexterous manipulation.
- We introduce EgoForce, a large-scale human tactile dexterous manipulation dataset and scalable collection platform for pretraining.
- Across ten real-world tasks, we show the benefits of force supervision in human pretraining data and visual track and future-force prediction during training, together with the effectiveness of FACT in using human data for fine-tuning.

II. Related Work

Large egocentric corpora [3, 4] have enabled robot learning from human experience through pretrained visual representations [5], imitation from human play [6], portable in-the-wild capture systems [7, 8], egocentric co-training [9, 10], and human–robot mixtures in VLA pretraining [11]. These sources are largely force-blind, capturing hand motion without contact forces. *EgoForce* (Section III) closes this gap with 2,000 hours of human manipulation paired with calibrated tactile force.

Learning from this experience requires policies that use touch to guide dexterous execution. Tactile sensing has supported robot manipulation through vision-based skins [12, 13], learned touch representations [14, 15], in-hand control [16], and tactile-aware imitation and VLA policies [17, 18]. Related work has explored force-centric demonstration capture [19] and auxiliary future contact or force prediction [20]. Hierarchical manipulation further separates planning from fine-grained control through explicit action hierarchies [21, 22], staged VLA post-training [23], and slow-fast tactile refinement [11, 24]. T-Rex [11] combines low-frequency visuomotor action generation with asynchronous tactile refinement, conditioning coarse actions on vector-quantized force history [25]. Force remains an input, leaving action–contact dynamics implicit and refinement dependent on the contact information preserved by compression. FACT instead jointly denoises continuous future force and action, allowing them to shape one another during refinement.

Future video and visual track prediction support human-to-robot transfer through a shared representation of anticipated scene motion across different action spaces. Work coupling action generation with future visual prediction includes video pretraining for manipulation [26], policy-as-video [27], and interactive world simulators [28, 29]. Visual point tracks provide an explicit, embodiment-agnostic description of scene motion. They have served as policy interfaces [30], transfer media from human video [31–33], and cross-domain motion representations [32], enabled by general-purpose trackers [34]. Similar to [33], FACT jointly predicts actions, future video, and visual tracks in its Wan2.2-based coarse planner [35]. These shared prediction targets let human demonstrations supervise motion representations that ground robot action generation in anticipated visual motion.

III. The EgoForce Dataset

Human demonstrations provide a scalable source of dexterous manipulation experience, but existing human data captures vision without the forces involved in interaction, while collecting force-supervised data at comparable scale on robots is expensive and difficult. We therefore develop *EgoForce*, a large-scale human tactile dataset paired with a collection platform that captures synchronized vision, hand motion, and calibrated tactile force for FACT pretraining.

a) Scalable multimodal collection platform: We design a wearable platform to capture natural bimanual manipulation while minimally constraining the demonstrator. Each participant wears a pair of tactile gloves together with three egocentric RGB cameras mounted on the head and both wrists. The platform synchronizes the visual streams with glove motion and tactile force measurements, producing temporally aligned multimodal trajectories across diverse tasks and environments.

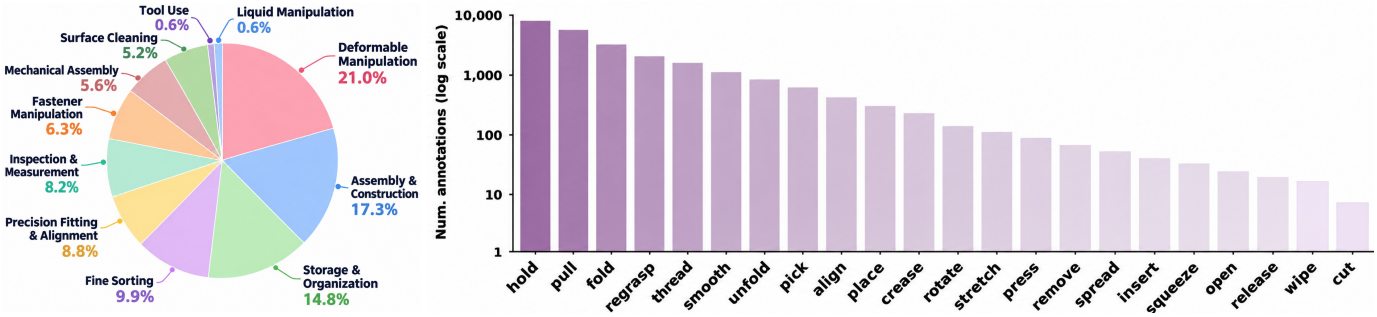


Fig. 2: ***EgoForce* task and skill distribution.** Left: episode share by task category, with over two thirds of the corpus covering contact-rich fine manipulation across diverse task categories. Right: atomic-skill annotations on a log scale, showing a broad distribution of dexterous, force-dependent skills beyond simple transport.

b) Automatic force calibration: To obtain comparable force measurements across sensing elements, gloves, and collection sessions, we develop an automatic calibration suite that maps raw A/D readings to physical force. A pneumatic bladder applies approximately uniform pressure across the tactile surface, swept from 0 to 50 psi, while a reference pressure sensor records pressure synchronously with glove readings. For each sensing element, we compute force as $F_i = P_i A_i$, where P_i is the reference pressure and A_i the sensing area. These paired A/D and force measurements yield a per-element calibration mapping applied during data collection.

c) The EgoForce dataset: Using this platform, we collect *EgoForce*, comprising approximately 2,000 hours of human demonstrations across 40k episodes, 1,000 tasks, 120 scenes, and 1,000 demonstrators. Each demonstration contains synchronized multi-view egocentric RGB observations, human hand motion, and calibrated tactile force signals, capturing both manipulation behavior and the physical contact underlying it.

d) Task distribution: dexterous by construction: Figure 2 shows that *EgoForce* is deliberately skewed toward dexterous, contact-rich interaction rather than pick-and-place transport. The largest categories are deformable manipulation (21.0%), assembly & construction (17.3%), and storage & organization (14.8%); together with fine sorting, precision fitting, fastener manipulation, and mechanical assembly, fine manipulation comprises over two thirds of the data. At the verb level, after *hold* and *pull*, frequent annotations include *fold*, *regrasp*, *thread*, *smooth*, and *crease*, reflecting in-hand reorientation and deformable skills whose success depends on contact forces, while *pick* and *place* rank only eighth and tenth. Similar visual states, such as a held fold or threaded lace, can require very different force profiles, motivating calibrated force supervision.

IV. The FACT Framework

FACT is a flow-matching model for transferring dexterous manipulation from large-scale human experience to robot execution. FACT cascades from prospective visual planning to force-aware refinement (Figure 3) through two jointly trained components. The Track-Guided Coarse Planner jointly predicts future video, visual tracks, and actions. Future video and tracks provide a shared representation of task progress across human and robot embodiments, allowing human experience to shape coarse robot motion. The Force–Action Joint Refiner jointly refines the action suffix and predicts future fingertip force while cross-attending to the coarse representation from the Track-Guided Coarse Planner. This separation supports cross-embodiment learning through shared visual motion and fine-grained robot execution through force–action co-prediction. We pretrain FACT on our *EgoForce* dataset together with 100 hours of robot teleoperation demonstrations from T-Rex [11].

At inference time, the policy receives the current visual observation, robot state, task instruction, visual track history, and tactile history. Given these observations, FACT outputs a 16-step, 20 Hz action chunk $a \in \mathbb{R}^{16\times 62}$, consisting of delta-base pose and absolute hand joints for each arm, and the prediction of future force at the chunk's 5 Hz refresh slots, $\hat{F} \in \mathbb{R}^{4\times 10\times 6}$, as well as the future visual track and video prediction.

A. Track-Guided Coarse Planner

The Track-Guided Coarse Planner learns a coarse visuomotor plan by jointly predicting future video, visual tracks, and actions, which is instantiated with a block-causal video DiT that autoregressively models the interaction over time.

a) Multimodal representation: As illustrated in Figure 3, the Track-Guided Coarse Planner separates its inputs into three groups. Task instruction and the anchor frame, encoded by umT5-XXL [36] and CLIP [37], provide cross-attention conditioning. Robot state, past visual tracks, and conditioning video latents provide clean context, while future video latents, actions, and future visual tracks are jointly denoised. The model operates autoregressively over temporal blocks containing video, action, state, and past and future visual-track tokens. Additional clean observation tokens condition the autoregressive sequence, while text and CLIP features remain outside the self-attention sequence and enter only through cross-attention.

b) Per-token flow matching: The Track-Guided Coarse Planner jointly generates future video, actions, and tracks through flow matching. For each generated modality with clean target c and Gaussian noise ε, we construct

$$x_\tau = (1-\tau)c + \tau\varepsilon, \qquad v^* = \varepsilon - c, \tag{1}$$

and train the model to predict the corresponding velocity field.

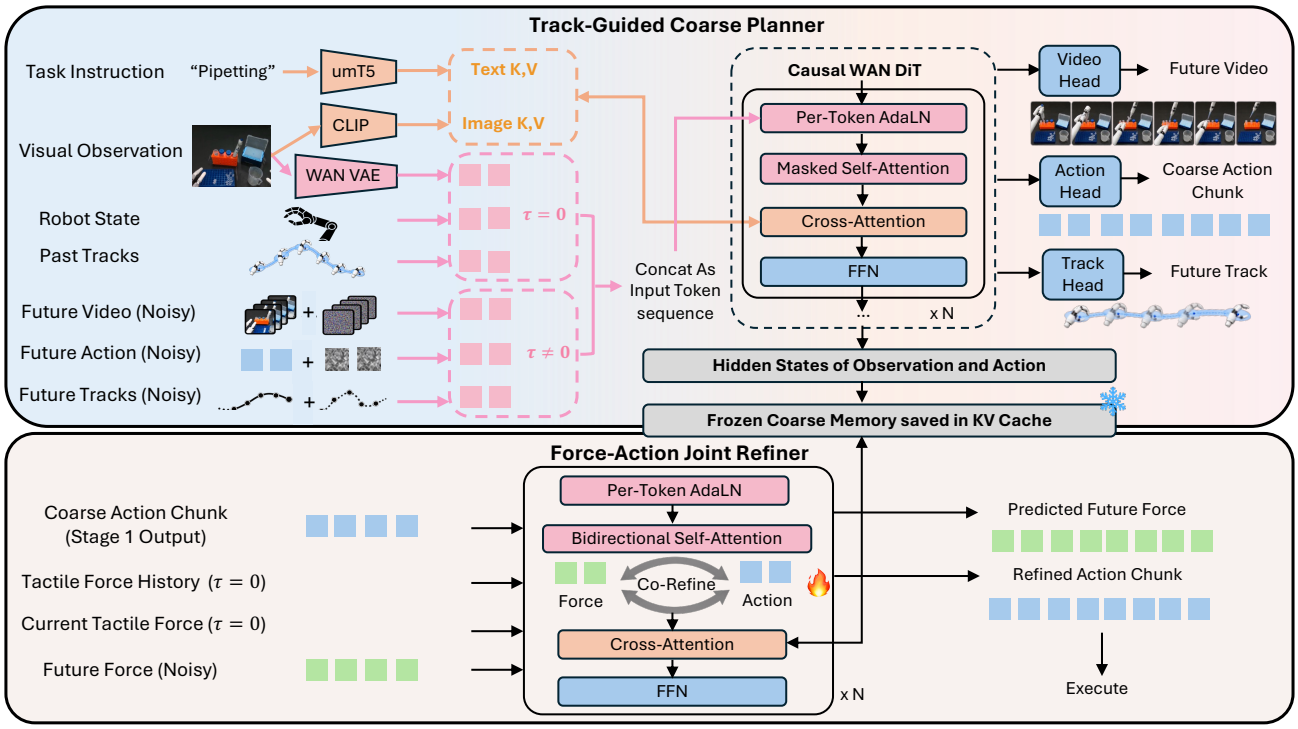


Fig. 3: **FACT architecture.** The Track-Guided Coarse Planner jointly denoises future video, visual tracks, and a coarse action chunk with a block-causal video DiT; per-token noise conditioning lets clean context and noisy targets coexist. At the τ=0.4 handoff, video and tracks are read out and the coarse hidden states are frozen. The Force–Action Joint Refiner reads them through cached cross-attention and co-denoises the remaining action ($\tau_a : 0.4 \to 0$) with continuous future fingertip force ($\tau_f : 1 \to 0$), using observed tactile signals as clean context.

Each token is conditioned on its own noise level τ, with τ=0 denoting clean data and τ=1 pure noise. Sinusoidal embeddings of τ are projected to six AdaLN modulation vectors per DiT block. Conditioning latents, state, and past tracks remain at τ=0, while future video latents, actions, and tracks use block-wise sampled noise levels $\tau(k) \sim \mathrm{Beta}(1.5, 1)$. This per-token conditioning lets clean context and noisy targets coexist in one forward pass, enabling Track-Guided Coarse Planner to stop action denoising at an intermediate noise level and Force–Action Joint Refiner to continue refinement.

c) Attention topology: Each DiT block applies per-token AdaLN, masked self-attention, cross-attention, and a feed-forward network. Within a four-block causal window, observation and action tokens attend to visual observation, action, and track tokens in the current and preceding blocks, plus the current state token. State tokens remain local to their block; track tokens attend only to tracks, preserving a dedicated geometric pathway. Cross-attention incorporates text and CLIP features, and modality-specific heads decode future video, the coarse action chunk, and future tracks.

d) Training objective: The Track-Guided Coarse Planner jointly optimizes visual dynamics, action, and track prediction:

$$\mathcal{L}_{\text{coarse}} = \mathcal{L}_{\text{dyn}} + \mathcal{L}_{\text{act}} + \lambda_t \left(\mathcal{L}_{\text{trk}}^{\text{flow}} + 5\,\mathcal{L}_{\text{trk}}^{\text{pos}} + 2\,\mathcal{L}_{\text{trk}}^{\text{smooth}} \right), \tag{2}$$

where λ_t=20. Track losses are visibility-masked with a motion-weighted boost to emphasize informative trajectories. In addition to the track flow objective, the position and smoothness terms are computed from the one-step reconstruction

$$\hat{x}_0 = x_\tau - \tau \hat{v}. \tag{3}$$

e) Coarse handoff to the Force–Action Joint Refiner: At inference, the coarse planner jointly denoises future video, tracks, and action for six Euler steps, stopping at the intermediate noise level τ=0.4. Future video and tracks are then decoded in closed form, while the action remains partially denoised at τ_a=0.4 and is passed to the Force–Action Joint Refiner for further force-aware refinement. The frozen hidden states of visual observation and action tokens form the coarse representation passed to the refiner, where they are projected once and stored as a cross-attention key–value cache.

B. Force–Action Joint Refiner

The Force–Action Joint Refiner jointly refines the remaining action and predicts future fingertip force. At the coarse handoff, future video and tracks are read out, while the action remains partially denoised at τ_a=0.4. Observed tactile signals provide the current contact context, while future force is explicitly predicted alongside the action, allowing force and action to reciprocally shape one another throughout refinement.

a) Force–action joint refinement: The defining choice of the Force–Action Joint Refiner is that continuous future force is a *co-denoised prediction target, not an observation*: observed fingertip force enters as clean context, while the

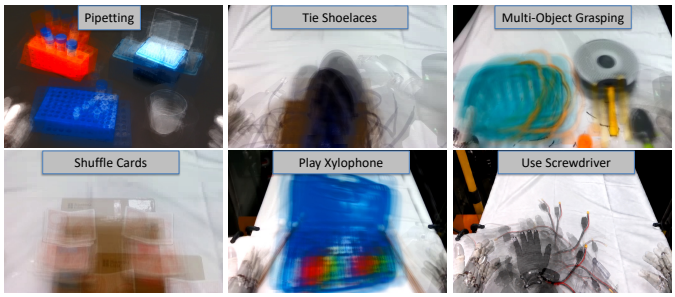


Fig. 4: **Evaluation randomization.** Overlaid initial frames from six evaluation tasks illustrate rollout-level variation in object positions and poses.

future per-fingertip force trajectory starts from pure noise at $\tau_f{=}1$ and is denoised jointly with the action suffix—unlike T-Rex [11], where force reaches the refiner only as vector-quantized observation codes and is never predicted. Each refiner block applies per-token AdaLN(τ), bidirectional self-attention, cross-attention to the coarse memory, and a feed-forward network; bidirectional attention between the action suffix and the noisy future-force tokens lets predicted force shape the evolving action and the action shape its predicted force consequences at every sampling step. Current tactile measurements thus describe the ongoing physical interaction, while predicted future force represents the physical consequence of the action being refined.

b) Coarse memory through cached cross-attention: The frozen coarse hidden states provide the coarse visual plan through cross-attention. At the $\tau{=}0.4$ handoff, the hidden states of observation and action tokens are detached and retained, while track, state, and conditioning tokens are dropped. These hidden states are projected once and stored as cross-attention key–values. Because the coarse representation remains fixed after the handoff, the cached key–values are reused across all four Euler steps and four tactile refreshes. This allows repeated force–action refinement against the same coarse visual plan without recomputing the Track-Guided Coarse Planner.

c) Cascaded sampling: Sampling performs four joint Euler steps with asymmetric schedules $\tau_a : 0.4{\to}0$ and $\tau_f : 1{\to}0$. At each step, only the remaining action suffix $a_{[o:16)}$ and unobserved future-force slots are updated; the executed action prefix and observed force slots remain fixed. At the next tactile refresh, the newly observed interaction is incorporated and the remaining action suffix is refined again.

d) Training objective: The Force–Action Joint Refiner jointly optimizes action refinement and future-force prediction:

$$\mathcal{L}_{\text{refine}} = \underbrace{\left\|\hat{v}_a - v_a^*\right\|^2_{[o:16)}}_{\text{action suffix}} + \lambda_f \underbrace{\left\|\hat{v}_F - v_F^*\right\|^2_{s>o/4}}_{\text{future force}} + \mathcal{L}_{\text{VQ}}. \quad (4)$$

We sample $\tau_a \sim \text{Beta}(1.5, 1)\cdot 0.4$ and $\tau_f \sim \text{Beta}(1.5, 1)$. The coarse memory forward is evaluated at exactly $\tau{=}0.4$ using the same suffix noise draw, matching the cascaded inference state during training. We use $\lambda_f{=}0.5$ in all experiments.

TABLE I: Real-robot evaluation on the ten-task suite (scoring in Section V-A). For pipette volume, the indented second row gives the mean absolute error in μL over the engaged trials. *w/o force pred* and *w/o track pred* remove future-force prediction and visual track prediction during training, respectively.

#	Task	$\pi_{0.5}$	T-Rex	FACT (ours) w/o force pred	FACT (ours) w/o track pred	FACT (ours) full
Binary tasks: success rate (%) ↑						
1	Lab pipetting	0	0	30	20	**40**
2	Open test-tube cap	10	20	40	**80**	**80**
3	Open microcentrifuge flip cap	0	20	20	40	**50**
4	Tie shoelaces	0	0	0	**20**	**20**
5	Shuffle cards	0	10	0	20	**30**
6	Multi-object grasp	0	0	10	**40**	**40**
7	Use screwdriver	0	10	0	**30**	**30**
	Mean	1.4	8.6	14.3	35.7	**41.4**
Continuous tasks: clean notes of 10 ↑						
8	Play harp	0.5	2.1	0	6.5	**6.7**
9	Play xylophone	0	3.2	1.9	3.9	**5.4**
Volume adjustment: engaged trials of 10 ↑						
10	Pipette volume	1	2	2	4	**5**
	mean abs. error (μL) ↓	17.0	14.0	11.0	12.8	**6.4**

C. Implementation Details

The Track-Guided Coarse Planner is a 5B video DiT based on Wan2.2-TI2V (30 blocks, $d = 3072$, 24 heads). Each 0.8 s block spans two latent video frames and contains 617 tokens: 100 video, 16 action, one state, and 250 each for past and future tracks; 50 clean observation tokens are prepended once. Video uses factorized 3D RoPE, while action, state, and tracks use separate 1D RoPE. Tracks additionally encode view, hand, semantic role, point, block, and past/future identity. The Force–Action Joint Refiner is a 6-block bidirectional transformer ($d = 768$, 12 heads, FFN $d = 3072$), with cached coarse states projected from 3072 to 768 for cross-attention. All components, including the track codec, tactile VQ module, and modality embeddings, are trained from scratch end to end. Training uses $K = 4$ teacher-forced blocks (3.2 s); inference generates one block at a time autoregressively. Tactile dropout zeros all tactile tokens with probability 0.1 per sample.

D. Training Recipe

The Track-Guided Coarse Planner and the Force–Action Joint Refiner are jointly trained by:

$$\mathcal{L}_{\text{FACT}} = \mathcal{L}_{\text{coarse}} + \lambda_{\text{refine}}\mathcal{L}_{\text{refine}}, \quad (5)$$

We pretrain FACT on a mixture of 2,000 hours of *EgoForce* human demonstrations (Section III) and 100 hours of T-Rex robot teleoperation data [11], sampled proportionally to hours, on 240 Alibaba T-Head PPU accelerators (Zhenwu 810E, 96 GB HBM2e; Table IV). The entire model is trained end to end under the joint objective above: the full DiT backbone is fine-tuned from Wan2.2-TI2V-5B, and the track codec, joint refiner, tactile VQ, and modality embeddings are trained from scratch. Unlike a vision-only pretraining stage, the calibrated glove forces supervise the force pathway from the first step, so contact behavior is learned jointly with visuomotor priors rather than grafted on afterwards.

TABLE II: Pretraining recipe ablation on all ten tasks.

#	Task	human w/ force+robot (full)	human w/o force+robot	robot only
Binary tasks: success rate (%) ↑				
1	Lab pipetting	**40**	20	0
2	Open test-tube cap	**80**	0	20
3	Open microcentrifuge flip cap	**50**	10	0
4	Tie shoelaces	**20**	0	0
5	Shuffle cards	**30**	10	0
6	Multi-object grasp	**40**	**40**	30
7	Use screwdriver	**30**	10	0
Continuous tasks: clean notes of 10 ↑				
8	Play harp	**6.7**	2.0	1.2
9	Play xylophone	**5.4**	2.2	1.0
Volume adjustment: engaged trials of 10 ↑				
10	Pipette volume	**5**	2	0
	mean abs. error (μL) ↓	**6.4**	20.5	–

V. Experiments

A. Experimental Setup

a) Robot platform and data collection: We evaluate FACT on a ten-task real-robot suite. All policies are evaluated on a bimanual YAM station with two Sharpa Wave hands (22 joints, five-fingertip 6-axis force sensing), equipped with a RealSense D435 ego-view camera and two ZED Mini wrist cameras. For each evaluation task we collect 10 robot teleoperation trajectories using a Manus-glove and wrist-tracker interface and 90 human demonstrations collected with a Wuji tactile glove and wrist-tracker interface, resulting in a 100-trajectory mixture dataset per task.

b) Skill-specific fine-tuning: We fine-tune the pretrained FACT on the 100-trajectory dataset for each task on 8 GPUs. This preserves the force-reactive behaviors acquired in pretraining while adapting to specific tasks.

c) Baselines and ablations: We compare with two baselines, $\pi_{0.5}$ [38] and T-Rex [11], each fine-tuned from its public base checkpoint on the same dataset. We additionally compare two FACT ablations: *w/o force pred* and *w/o track pred*, which remove future-force prediction and visual track prediction during training, respectively.

d) Evaluation protocol and metrics: The suite covers ten contact-rich tasks in three scoring classes. *Binary tasks* (seven: lab pipetting, test-tube cap opening, microcentrifuge flip-cap opening, shoelace tying, card shuffling, multi-object grasping, screwdriver use) score success over ten rollouts. *Continuous tasks* (two) pluck each of ten indicated harp strings or strike each of ten indicated xylophone bars and score the number of notes that sound cleanly. *Volume adjustment* starts the pipette dial at 420 μL with a 400 μL target; we report the number of trials (of ten) that *engage* the dial and, in an indented second row, the mean absolute error of the final reading over engaged trials. Reporting both prevents rare engagement from appearing precise; a never-engaged trial is a failure with no reading. All methods share the same scenes and objects across rollouts, with object placement randomized per rollout (Figure 4).

TABLE III: Contribution of human fine-tuning data on three dexterous tasks: ten robot trajectories alone (10:0) versus the 10:90 robot-to-human mixture, scored as in Table I.

		robot : human data	
#	Task	10 : 0	10 : 90
Binary tasks: success rate (%) ↑			
1	Tie shoelaces	10	**20**
2	Shuffle cards	10	**30**
Volume adjustment: engaged trials of 10 ↑			
3	Pipette volume	4	**5**
	mean abs. error (μL) ↓	11.2	**6.4**

B. Main Results

Table I shows that FACT consistently outperforms the baselines across the ten-task suite. On representative binary tasks, it achieves 40/80/40% success on pipetting, test-tube cap opening, and multi-object grasping, compared with 0/20/0% for T-Rex and 0/10/0% for $\pi_{0.5}$. It also sounds 6.7 of 10 harp notes cleanly, compared with 2.1 for T-Rex and 0.5 for $\pi_{0.5}$. The difference is especially pronounced on force-critical tasks: FACT engages the pipette-volume dial in 5 of 10 trials with 6.4 μL mean error, compared with 2 engagements and 14 μL error for T-Rex. Among the five successful engagements, four finish within 4 μL of the target, suggesting that failures arise primarily from engaging the dial rather than adjusting it once engaged. Together with the harp results, these results demonstrate FACT's ability to translate force-rich human pretraining and joint continuous force–action denoising into precise, contact-sensitive robot behavior. Figure 5 illustrates this behavior on three representative tasks: shoelace tying, where both hands pinch, cross, and pull the laces with coordinated multi-finger regrasps; pipette-volume adjustment, where the fingertips turn the dial in small increments while the rest of the hand holds the pipette steady; and multi-object grasping, where the hand secures a second object without releasing the first. In each case success is decided by fingertip contact rather than gross arm motion.

C. Ablation Studies

a) Force and track prediction: Table I shows the effectiveness of future-force and visual track prediction during training. Removing force prediction causes substantial degradation on contact-sensitive tasks, including test-tube cap opening ($80 \rightarrow 40\%$), harp playing ($6.7 \rightarrow 0$ notes), and pipette-volume adjustment ($5/10 \rightarrow 2/10$ engagements). Removing track prediction produces smaller but consistent drops on tasks requiring precise motion, including pipetting ($40 \rightarrow 20\%$), card shuffling ($30 \rightarrow 20\%$), and xylophone playing ($5.4 \rightarrow 3.9$ notes). Together, these results show force prediction primarily supports contact-rich execution, while tracks improve precise motion.

b) Force supervision in human pretraining data: To isolate the contribution of force-supervised human data, we compare three pretraining recipes across the ten tasks (Table II): human demonstrations with glove-measured force mixed with robot teleoperation, the same mixture without human force supervision, and robot-only training. Removing force supervision causes the largest drops on contact-sensitive tasks: test-tube

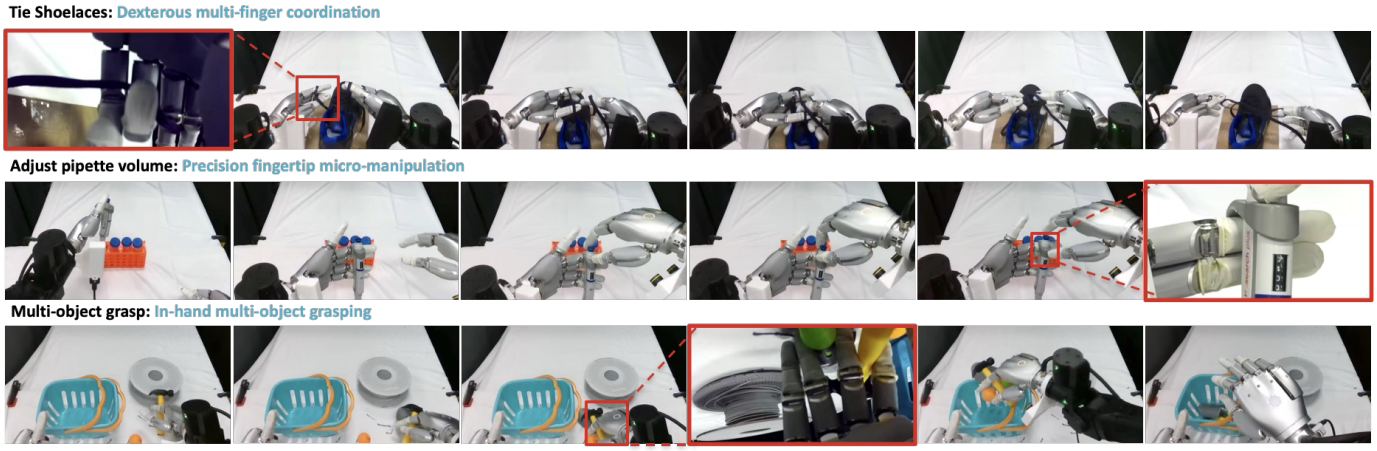


Fig. 5: **Qualitative examples.** FACT executes contact-rich behaviors requiring dexterous multi-finger coordination, precision fingertip micro-manipulation, and in-hand multi-object grasping.

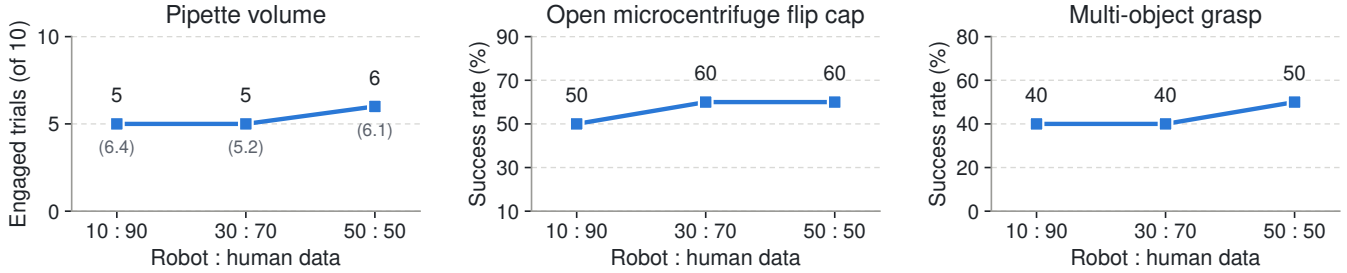


Fig. 6: **Robot-to-human mixture in fine-tuning.** Performance versus the robot-to-human ratio of the 100-trajectory per-task fine-tuning mixture (10:90, 30:70, 50:50).

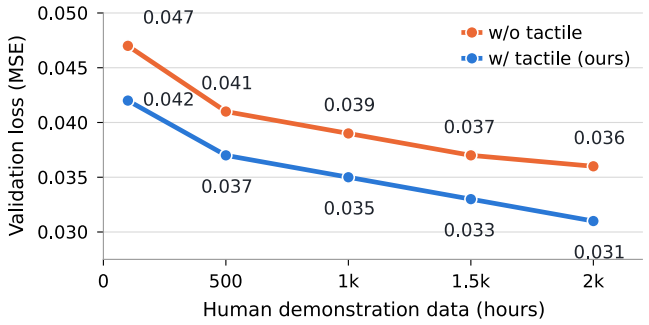


Fig. 7: **Scaling with human data.** Validation loss (MSE) versus hours of human demonstration data (100, 500, 1k, 1.5k, and 2k hours on a linear axis), mixed with robot teleoperation data, with and without tactile supervision. Tactile supervision lowers the loss at every scale and the gap persists as data grows.

cap opening falls from 80% to 0%, pipetting from 40% to 20%, pipette-dial engagement from 5/10 to 2/10 with mean error increasing from 6.4 to 20.5 μL, and xylophone performance from 5.4 to 2.2 clean notes. In contrast, multi-object grasping remains unchanged at 40%. Removing human data entirely also degrades performance, yielding 0% on pipetting and flip-cap opening, 1.2 clean harp notes, and no successful dial engagement. These results show that FACT benefits substantially from human demonstrations during pretraining, with force supervision crucial to translating this experience into strong performance on robot dexterous manipulation tasks.

c) Scaling with pretraining human data: To test how downstream robot policy performance scales with pretraining human data, we train FACT with 100, 500, 1k, 1.and 2k hours of human demonstrations, each mixed with the same robot teleoperation data, and compare training with and without tactile supervision. Figure 7 shows that validation loss decreases log-linearly with human data in both settings, while tactile supervision provides a consistent 10–14% improvement at every scale (0.047→0.042 at 100 h and 0.036→0.031 at 2k h). The benefit persists as data scales: from 1k to 2k hours, validation loss decreases by 11% with tactile supervision, compared with 8% without it. All models are trained on 240 Alibaba T-Head PPUs.

d) Human data in fine-tuning: Table III isolates the contribution of human data on three dexterous tasks by comparing fine-tuning with ten robot trajectories alone (10:0) against the full 10:90 robot–human mixture. Adding the 90 human demonstrations improves shoelace tying from 10% to 20%, card shuffling from 10% to 30%, and pipette-dial engagement from 4/10 to 5/10 while reducing mean error from 11.2 to 6.4 μL. These gains demonstrate FACT's ability to transfer manipulation knowledge from human demonstrations into improved robot task completion and precision, making effective use of human experience when robot data is limited.

e) Human-demonstration efficiency in fine-tuning: Figure 6 examines how performance changes with the robot-to-human data ratio in the 100-trajectory per-task fine-tuning

mixture, using robot-to-human ratios of 10:90, 30:70, and 50:50. Increasing robot data yields modest gains: flip-cap opening improves from 50% to 60%, multi-object grasping from 40% to 50%, and pipette-dial engagement from 5/10 to 6/10. Pipette-volume error remains similar across the three mixtures (6.4, 5.2, and 6.1 μL). These results suggest that FACT learns from human demonstrations nearly as effectively as from robot demonstrations for fine-tuning.

VI. Conclusion

We present FACT, a dexterous robot foundation model with force-aware human pretraining. By combining large-scale force-rich human experience with robot demonstrations, FACT couples prospective visual planning with force-aware action refinement: future video and visual tracks guide coarse motion, while predicted future force shapes fine-grained execution. Across ten dexterous tasks, our experiments demonstrate the complementary roles of tracks and force prediction and show that human demonstrations can substantially reduce the amount of task-specific robot data required for fine-tuning.

References

[1] E. Todorov and M. I. Jordan, "Optimal feedback control as a theory of motor coordination," *Nature Neuroscience*, vol. 5, no. 11, pp. 1226–1235, 2002.

[2] S. H. Scott, "Optimal feedback control and the neural basis of volitional motor control," *Nature Reviews Neuroscience*, vol. 5, no. 7, pp. 532–545, 2004.

[3] K. Grauman *et al.*, "Ego4D: Around the world in 3,000 hours of egocentric video," in *CVPR*, 2022.

[4] D. Damen *et al.*, "Scaling egocentric vision: The EPIC-KITCHENS dataset," in *ECCV*, 2018.

[5] S. Nair *et al.*, "R3M: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.

[6] C. Wang *et al.*, "MimicPlay: Long-horizon imitation learning by watching human play," *Conference on Robot Learning (CoRL)*, 2023.

[7] C. Chi *et al.*, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots," *Robotics: Science and Systems (RSS)*, 2024.

[8] C. Wang *et al.*, "DexCap: Scalable and portable mocap data collection system for dexterous manipulation," *arXiv preprint arXiv:2403.07788*, 2024.

[9] S. Kareer *et al.*, "EgoMimic: Scaling imitation learning via egocentric video," *arXiv preprint arXiv:2410.24221*, 2024.

[10] R. Hoque *et al.*, "EgoDex: Learning dexterous manipulation from large-scale egocentric video," *arXiv preprint arXiv:2505.11709*, 2025.

[11] D. Niu *et al.*, "T-Rex: Tactile-reactive dexterous manipulation," 2026.

[12] W. Yuan, S. Dong, and E. H. Adelson, "GelSight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, 2017.

[13] B. Wu *et al.*, "3D-ViTac: Learning fine-grained manipulation with visuo-tactile sensing," *arXiv preprint arXiv:2410.24091*, 2024.

[14] C. Higuera *et al.*, "Sparsh: Self-supervised touch representations for vision-based tactile sensing," *Conference on Robot Learning (CoRL)*, 2024.

[15] I. Guzey *et al.*, "Dexterity from touch: Self-supervised pre-training of tactile representations," *CoRL*, 2023.

[16] H. Qi *et al.*, "General in-hand object rotation with vision and touch," *CoRL*, 2023.

[17] L. Heng *et al.*, "ViTacFormer: Learning cross-modal representation for visuo-tactile dexterous manipulation," *arXiv preprint*, 2025.

[18] J. Huang *et al.*, "Tactile-VLA: Unlocking vision-language-action model's physical knowledge for tactile generalization," *arXiv preprint arXiv:2507.09160*, 2025.

[19] W. Hu *et al.*, "ForceMimic: Force-centric imitation learning with force-motion capture system for contact-rich manipulation," *arXiv preprint arXiv:2410.07554*, 2024.

[20] H. Li *et al.*, "Force-aware imitation learning for contact-rich manipulation," *arXiv preprint*, 2024.

[21] S. Belkhale, Y. Cui, and D. Sadigh, "HYDRA: Hybrid robot actions for imitation learning," *CoRL*, 2023.

[22] S. Belkhale *et al.*, "RT-H: Action hierarchies using language," *arXiv preprint arXiv:2403.01823*, 2024.

[23] K. Black *et al.*, "$\pi^*_{0.6}$: A VLA that learns from experience," *arXiv preprint*, 2025.

[24] H. Xue *et al.*, "Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation," *arXiv preprint arXiv:2503.02881*, 2025.

[25] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," *NeurIPS*, 2017.

[26] H. Wu *et al.*, "Unleashing large-scale video generative pre-training for visual robot manipulation," *arXiv preprint arXiv:2312.13139*, 2023.

[27] Y. Du *et al.*, "Learning universal policies via text-guided video generation," *NeurIPS*, 2023.

[28] M. Yang *et al.*, "Learning interactive real-world simulators," *ICLR*, 2024.

[29] N. Agarwal *et al.*, "Cosmos world foundation model platform for physical AI," *arXiv:2501.03575*, 2025.

[30] C. Wen *et al.*, "Any-point trajectory modeling for policy learning," *RSS*, 2024.

[31] H. Bharadhwaj *et al.*, "Track2Act: Predicting point tracks from internet videos enables generalizable robot manipulation," *ECCV*, 2024.

[32] M. Xu *et al.*, "Flow as the cross-domain manipulation interface," *CoRL*, 2024.

[33] Z. Cao *et al.*, "TrAct: Bridging robot control and visual prediction with visual tracks," arXiv preprint arXiv:2608.24101, 2026.

[34] N. Karaev *et al.*, "CoTracker: It is better to track together," *ECCV*, 2024.

[35] Wan Team, "Wan: Open and advanced large-scale video generative models," *arXiv:2503.20314*, 2025.

[36] H. W. Chung *et al.*, "UniMax: Fairer and more effective

language sampling for large-scale multilingual pretraining," arXiv preprint arXiv:2304.09151, 2023.
[37] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," ICML, 2021.
[38] Physical Intelligence, K. Black *et al.*, "$\pi_{0.5}$: A vision-language-action model with open-world generalization," *arXiv preprint arXiv:2504.16054*, 2025.

ACKNOWLEDGEMENTS

The 2000 hours of vision-tactile sensing data used for model training in this work were collected with the FusionX multimodal data collection system (TouchTronix Robotics, NY, USA). The authors thank Dr. Jun Liu and Dr. Yixiang Gao of TouchTronix Robotics for fruitful scientific discussions on this manuscript.

The authors thank Sharpa and Xdof for the donation of the dexterous hands and robot arms used in the experiments.

APPENDIX A
MODEL AND TRAINING DETAILS

TABLE IV: Model and training configuration of FACT.

Track-Guided Coarse Planner (Flow Matching)	
Backbone	Wan2.2-TI2V-5B video DiT
Hidden dimension	3072
Number of layers	30
Max sequence length	2,518 tokens
Action dimension	62
Action chunk	16
Timestep sampling	$\mathrm{Beta}(1.5, 1.0)$
Num inference timesteps	6 (τ: 1→0.4)
Parameters	5.05B
Force–Action Joint Refiner (Flow Matching)	
Hidden dimension	768
Number of layers	6
FFN intermediate size	3072
Action dimension	62
Action chunk	16
Num inference timesteps	4
Parameters	65M
Training configuration	
Optimizer	AdamW
Peak learning rate	1×10^{-5}
Min learning rate	0
LR scheduler	cosine
Warmup ratio	0.05
Weight decay	10^{-5}
Gradient clipping	1.0
GPU type	Alibaba T-Head PPU (Zhenwu 810E, 96 GB HBM2e)
Number of GPUs	240
o DeepSpeed ZeRO stage	2
Per device batch size	16
Gradient accumulation steps	1
Mixed precision training	bf16
Tactile dropout	0.1

Table IV lists the configuration of both stages and the training setup. The Track-Guided Coarse Planner is the Wan2.2-TI2V-5B diffusion transformer fine-tuned in full; the action, state, track, and image projections, the modality-type embedding, and the entire Force–Action Joint Refiner are trained from scratch, while the VAE, the umT5-XXL text encoder, and the CLIP image encoder stay frozen.

a) Visual tracks: Tracks are extracted offline: SAM2 segments the hands in the first frame and CoTracker follows 250 points per block through the episode at 180×320 (50 points per hand in the head view; 50 hand points plus a 5×5 background grid in each wrist view), normalized to $[0, 1]$, with a 16-frame past horizon and a 16-step future horizon at 20 Hz. The track encoder flattens $[x, y, \mathrm{vis}, \Delta x, \Delta y]$ over 16 frames through $\mathrm{Linear}(80\to3072) \to \mathrm{SiLU} \to \mathrm{Linear}$ with view, hand, role, point, block, and past/future embeddings, and the decoder maps each future-track token to 16 two-dimensional velocities. Track losses are visibility-masked and weighted by $1 + 4\ell$, where ℓ is the point's path length.

b) Data pipeline: Demonstrations are stored in LeRobot-v2 format at 30 fps and resampled by nearest timestamp to 20 Hz for actions and tracks, 10 Hz for video, and 5 Hz for tactile force. The 62-D action concatenates, per arm, the end-effector position and 6-D rotation relative to the chunk's starting pose and 22 absolute hand joints; states, actions, and forces are normalized by their 1st/99th percentiles to $[-1, 1]$. The three camera views are resized to 160×320 and tiled 2×2 before VAE encoding, so that each latent frame yields 50 tokens.

c) Inference: The coarse stage runs six Euler steps from $\tau{=}1$ to 0.4 and one further forward pass at $\tau{=}0.4$ that reads out video and tracks and produces the hidden states cached as the refiner's cross-attention memory; the refiner then runs four joint Euler steps. No classifier-free guidance is used, and the resulting 16-step chunk is executed at 20 Hz.

APPENDIX B
FORCE–ACTION JOINT REFINER

a) Tactile input: Each hand provides five fingertip 6-axis wrenches $(F_x, F_y, F_z, M_x, M_y, M_z)$, giving a 60-D force observation per time step, normalized per channel by its 1st/99th percentiles. At each 5 Hz refresh slot the refiner receives the force history of the last 16 samples (3.2 s); missing samples at the start of an episode are zeroed and masked, and the current force is the last sample of the window.

b) Force-history quantization: The history of every fingertip is encoded by a 1D convolutional encoder shared across both hands: kernel-5 convolutions with GroupNorm and GELU, 128 channels, two stride-2 blocks, and a 256-D bottleneck that is mean-pooled over time into one 256-D code per fingertip. The codebook has 64 entries of 256 dimensions, updated by exponential moving average (decay 0.99) with dead codes re-seeded every 200 steps. The VQ loss is the reconstruction MSE plus a commitment term with weight 0.25, added to the total objective with weight 1; learned hand and finger embeddings are added to the resulting ten history tokens.

c) Token layout: A refiner instance operates on 76 tokens: 16 action tokens ($\mathrm{Linear}(64\to768)$ plus step-position, slot, type, and refresh-offset embeddings), 10 current-force tokens (a $6\to768\to768$ MLP plus finger and type embeddings), 10 VQ-history tokens ($\mathrm{Linear}(256\to768)$), and 40 future-force tokens for the four 5 Hz slots of the chunk, which reuse the force

MLP with finger and slot embeddings. Every block applies per-token AdaLN modulation (sinusoidal embedding of 1000τ through an MLP to six 768-D vectors, plus a learned per-block bias), full bidirectional self-attention over all 76 tokens, cross-attention to the coarse memory, and an AdaLN-modulated feed-forward network. The action head is a LayerNorm followed by $\mathrm{Linear}(768{\to}64)$ and the force head a LayerNorm followed by $\mathrm{Linear}(768{\to}6)$; both predict flow-matching velocities.

d) Coarse memory: The memory consists of the planner's hidden states for the observation and action tokens of the current block and up to four preceding blocks at $\tau{=}0.4$ (116 tokens for a single block), projected once through $\mathrm{Linear}(3072{\to}768)$ and a LayerNorm and reused across all refiner steps. During training the memory is produced by an extra forward pass of the planner at exactly $\tau{=}0.4$, without gradient, using the same suffix noise draw as the refiner so that training matches the cascaded inference state.

e) Training: For every training sample the refiner is instantiated for each of the four blocks and each refresh offset $o \in \{0, 4, 8, 12\}$, i.e. 16 instances per sample. The action prefix $a_{[0:o)}$ is clean, and the suffix receives $\tau_a = 0.4\,(0.999\,\mathrm{Beta}(1.5, 1) + 0.001)$; force slots at or before the offset are clean and later slots receive $\tau_f \sim \mathrm{Beta}(1.5, 1)$. The future-force targets are the normalized measured forces at the four slot times. The loss is the flow-matching error on the action suffix plus 0.5 times the error on the unobserved force slots, plus the VQ loss. Tactile dropout zeroes the current-force and history tokens with probability 0.1 per sample so the refiner cannot rely on touch alone.

f) Sampling and execution: At inference the refiner takes four joint Euler steps with τ_a on a linear grid from 0.4 to 0 and τ_f from 1 to 0, updating only the action suffix and the unobserved force slots while the executed prefix and observed slots stay fixed; observed slots are filled with the current measured force. The refined chunk is executed at 20 Hz. The coarse planner runs once per chunk; at each subsequent 5 Hz tactile refresh (offsets 4, 8, and 12 of the executing chunk) the refiner alone is re-run against the cached coarse memory, with the executed prefix and the newly measured force slots clamped, so the remaining suffix is re-refined with the latest contact information at a fraction of the planner's cost. The predicted future-force trajectory is recorded for analysis and is not used as a control reference.

Appendix C
Hardware and Data Collection

a) EgoForce collection (pretraining data): EgoForce is collected with the FusionX multimodal collection system (Figure 8). The operator wears a head-mounted stereo camera (Arducam AR0234 global-shutter pair, two $1920{\times}1200$ RGB imagers with a 60 mm baseline and factory-calibrated intrinsics and extrinsics, 30 fps, with a 200 Hz IMU), a monocular $1920{\times}1080$ RGB camera at 30 fps with an IMU on each wrist, and a pair of TouchTronix tactile gloves. Each glove carries a piezoresistive conductive-textile array with 12 taxels on every fingertip ($5{\times}12$ per hand) and 60 taxels on the palm, a bend sensor per finger, and an orientation IMU, all sampled at about 200 Hz; every sample stores the raw 8-bit A/D reading and the calibrated force in newtons of each taxel, the five bend values, and the hand orientation quaternion. Before a session, a per-user calibration records the resting A/D offset of every taxel and the minimum and maximum bend of every finger, which are subtracted and normalized at load time. All streams are time-stamped on a common clock and stored in one MCAP file per episode, so that any frame of the three videos can be paired with the tactile samples of the same instant. Taxel readings are converted to physical force with the pneumatic calibration rig of Section III: a uniformly pressurized chamber and a compliant bladder conform to the glove surface while a reference sensor records the applied pressure, giving a per-taxel pressure–A/D curve and the force $F_i{=}P_iA_i$ from the taxel's effective area.

b) Human action and force labels: Hand motion is not measured by motion capture but estimated from the head camera: ViTPose hand keypoints and HaMeR MANO hand meshes are fit to every frame of the left stereo view, temporally smoothed, and anchored to metric scale with the stereo depth, which yields the wrist pose and the finger joint angles of both hands in the head-camera frame. The glove's bend sensors and orientation IMU refine the finger flexion and hand orientation where the camera view of the hand is occluded. The finger joint angles are retargeted to the 22 joints of the Sharpa Wave hand and the wrist pose is expressed relative to the start of each action chunk, matching the delta-base convention of the robot action space (Section A). For force supervision, the 12 calibrated taxel forces of each fingertip are reduced to the 6-axis fingertip wrench used by the model: their sum gives the normal force, their force-weighted centroid gives the center of pressure, from which the two moments about the fingertip frame follow, and the tangential components and twisting moment, which a pressure array cannot observe, are set to zero. The resulting per-fingertip wrenches are resampled to the model's 5 Hz force slots and normalized with the same percentile statistics as the robot wrenches.

c) Robot platform: All rollouts and task-specific robot demonstrations use the bimanual station in Figure 9. Two I2RT YAM Ultra 6-DoF arms are mounted on a common base and driven by a YAM Box controller, which also carries the mast of the head camera. Each arm ends in a Sharpa Wave 1 dexterous hand with 22 joints, whose five fingertips each measure a 6-axis force/torque wrench, delivered by the hand's on-board inference at about 30 Hz. A RealSense D435 on the mast provides the head view and a ZED Mini on each wrist provides the two wrist views, all at $1280{\times}720$ and 30 fps. The policy observes the three views, the robot state, and the ten fingertip wrenches, and commands the 62-D action of Section A: a delta-base end-effector pose per arm, resolved to joint targets by inverse kinematics on the YAM model and tracked by the arms' built-in joint controllers, and 22 absolute joint targets per hand.

d) Teleoperation: Robot demonstrations are collected by an operator wearing Manus gloves with a VIVE Tracker 3.0 strapped to each wrist (Figure 9, bottom left). The glove's finger

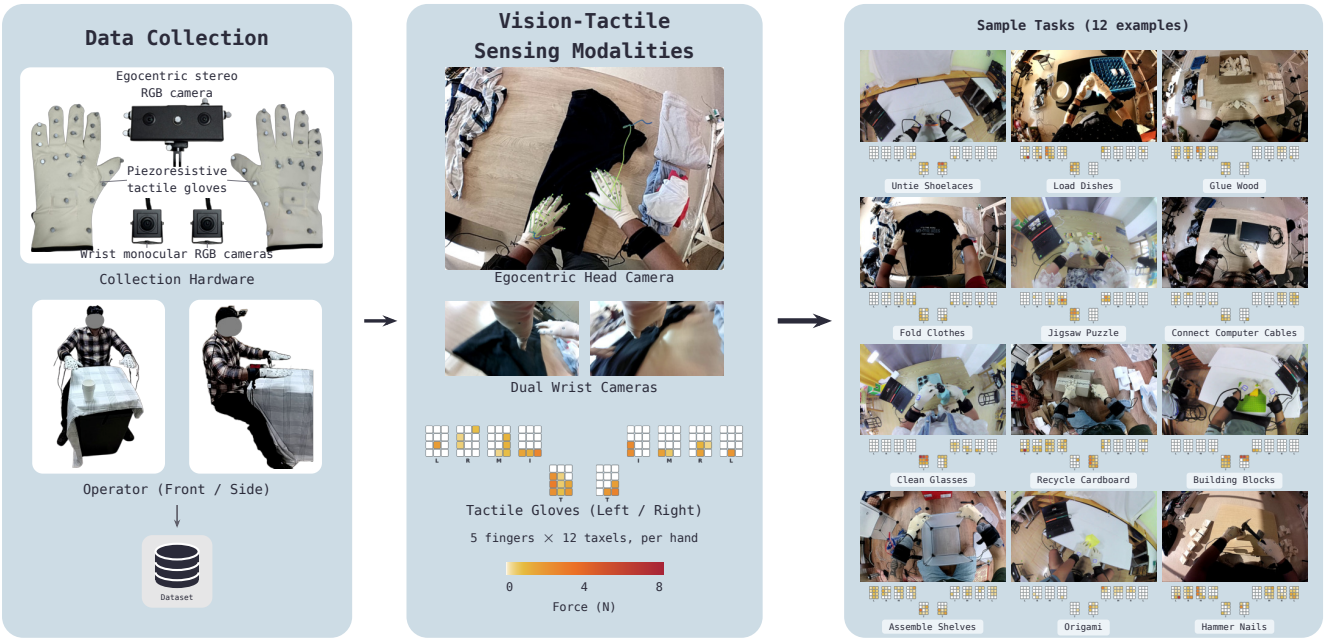


Fig. 8: **EgoForce collection system and data.** LEFT: collection hardware (egocentric stereo RGB camera, wrist RGB cameras, and piezoresistive tactile gloves) and front and side views of an operator wearing the system. MIDDLE: the FusionX multimodal collection system (TouchTronix Robotics, NY), illustrated with synchronized head and wrist views and tactile readings from a shirt-folding recording; the head view includes estimated hand poses and image-space hand trajectories, which are not motion-capture ground truth. RIGHT: twelve examples from the collection environments, each showing the egocentric view and the tactile maps of the left and right hand (finger labels T, I, M, R, L for thumb, index, middle, ring, and little finger). Heatmaps show per-taxel forces in newtons on a shared 0–8 N scale with square-root color mapping.

motion is retargeted to the 22 joints of the Sharpa Wave hand and sent to the hand driver, and the tracker pose is converted by inverse kinematics on the YAM model into arm joint targets; the operator watches the scene directly. A recorder running on the camera clock ticks at 30 Hz and, at every tick, stores the most recent commanded arm and hand targets, the measured arm poses and hand joints, and the latest tactile frame together with their wall-clock timestamps; the three video streams are hardware-timestamped and aligned to the same ticks, so that frame k of every stream and row k of the state log refer to the same instant. Demonstrations are stored in LeRobot-v2 format, and ten successful trajectories are kept per task.

e) Human demonstrations for fine-tuning: The 90 human demonstrations per task in the fine-tuning mixture are not taken from EgoForce but collected on the evaluation tasks themselves with a lighter interface (Figure 9, bottom right): the demonstrator wears Wuji tactile gloves, a RealSense D435 camera on each wrist that matches the robot's wrist views, and a VIVE Tracker 3.0 on each wrist. The trackers give the wrist poses, the gloves give the finger joint angles and fingertip tactile readings, and the wrist cameras give the two wrist views, all time-stamped on the same clock as the teleoperation recorder. The finger angles are retargeted to the 22 Sharpa Wave joints with the same mapping as in teleoperation, the wrist poses are expressed as delta-base end-effector poses, and the tactile readings are mapped to the per-fingertip wrench inputs of the model, so that a human demonstration enters fine-tuning in the same action and force space as a robot demonstration.

f) Robot pretraining data: The 100 hours of robot teleoperation data mixed into pretraining come from T-Rex [11], collected on a Dexmate Vega-1 bimanual platform (7-DoF arms) with the same Sharpa Wave hands and fingertip sensors, a ZED X Mini head camera, and two ZED X One S wrist cameras at 640×360, teleoperated with Manus gloves and VIVE trackers. Because arm actions are end-effector poses rather than joint angles, the same 62-D action space covers both arm embodiments.

APPENDIX D
IMPLEMENTATION DETAILS OF BASELINES

a) $\pi_{0.5}$: We use the official OpenPI implementation and the released $\pi_{0.5}$ base checkpoint, fine-tuned per task with full-parameter updates (no LoRA) under FSDP on 8 GPUs. The policy takes the three camera views (head, left wrist, right wrist), the task instruction, and the robot state; it has no tactile input. Because the $\pi_{0.5}$ action head is fixed at 32 dimensions, the bimanual action is encoded as the 6-D end-effector pose of each arm (position and roll–pitch–yaw, predicted as deltas) and a 10-D PCA projection of each hand's 22 joints (absolute; the basis retains 96–97% of the joint variance), and decoded back

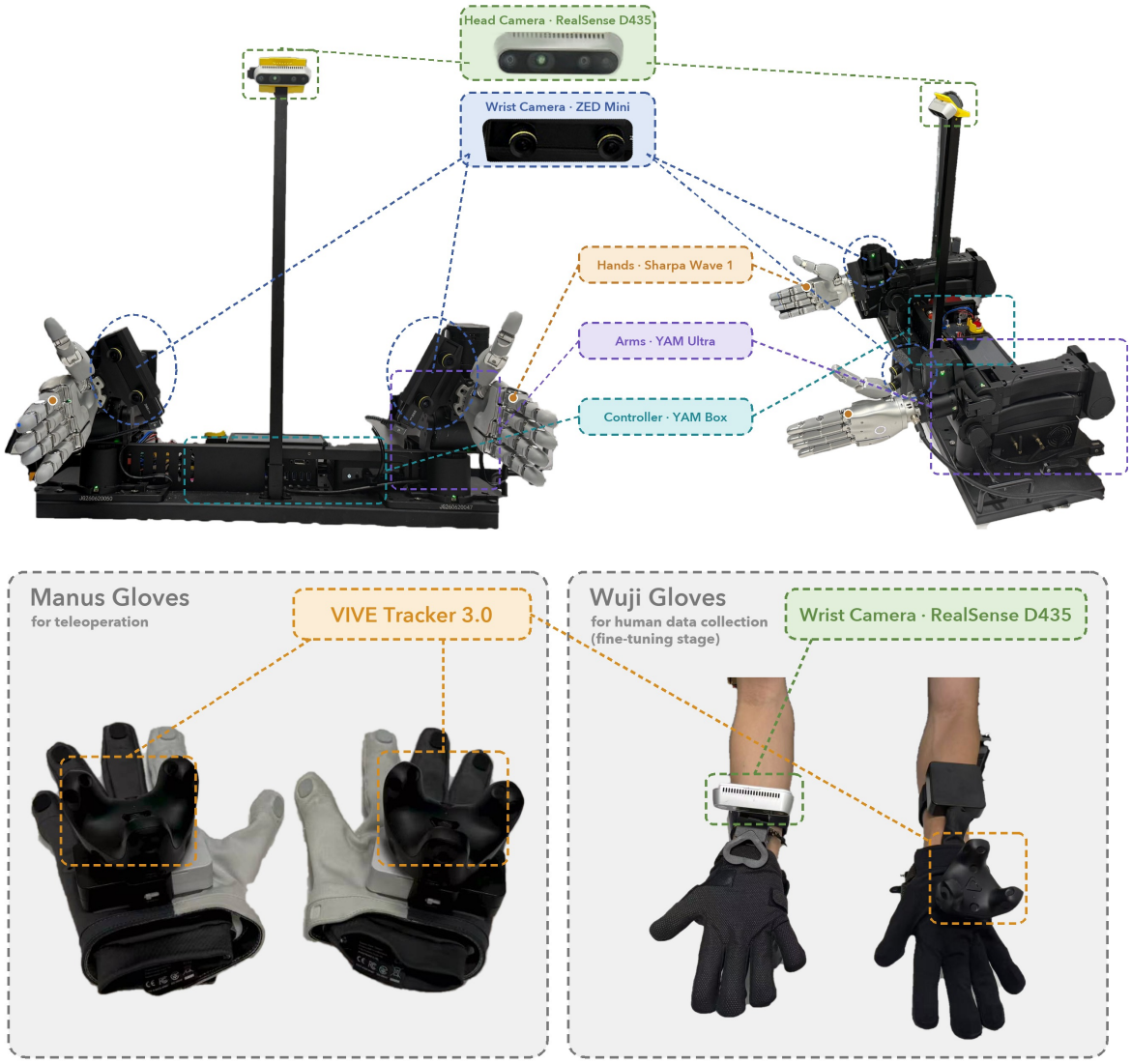


Fig. 9: **Real-world setup and data-collection interfaces.** Top: the bimanual station, with two I2RT YAM Ultra arms driven by a YAM Box controller, a Sharpa Wave 1 hand on each arm, a RealSense D435 head camera, and a ZED Mini camera on each wrist. Bottom left: the teleoperation interface, Manus gloves with a VIVE Tracker 3.0 on each wrist. Bottom right: the human demonstration interface used for the fine-tuning data, Wuji tactile gloves with a RealSense D435 wrist camera and a VIVE Tracker 3.0 on each wrist.

to 22 joints at deployment. Action horizon 10. Training uses AdamW with gradient clipping 1.0, a cosine schedule from a peak learning rate of 2.5×10^{-5} to 2.5×10^{-6} over 20k steps with 200 warmup steps, global batch size 32, and an EMA of the weights with decay 0.999.

b) T-Rex: We use the official T-Rex implementation and start from its released mid-trained checkpoint, which includes the tactile VQ-VAE and is the recommended starting point for post-training. T-Rex shares the action space of FACT: the same 62-D delta-base end-effector pose plus 22 hand joints per arm and the same 16-step chunk, so our demonstrations are used without any re-encoding. The model takes the three camera views at 384×288, the instruction, and the 6-D fingertip wrenches of all ten fingers through its VQ tactile encoder; the deformation-image branch is disabled, since our hands provide no deformation images, and the FLARE visual-prediction objective is kept on. Cascaded denoising uses 10 steps split at step 6 with tactile dropout 0.1. Training uses AdamW with a peak learning rate of 1×10^{-4}, cosine decay to zero without warmup, weight decay 0, gradient clipping 1.0, per-device batch size 8 on 8 GPUs with DeepSpeed ZeRO-2 in bf16, for 30 epochs.

c) FACT ablations: w/o force pred removes the future-force tokens and their loss from the Force–Action Joint Refiner, leaving the refiner to denoise the action suffix from the observed tactile context only. *w/o track pred* removes the track tokens,

the track heads, and the track losses from the Track-Guided Coarse Planner, so coarse actions are generated from video and state alone. Both variants are pretrained and fine-tuned with the recipe of Table IV. The pretraining-recipe variants of Table II keep the architecture fixed: *human w/o force* trains on the same human–robot mixture with the human force supervision disabled and the human tactile inputs masked, and *robot only* pretrains on the T-Rex robot data alone.

APPENDIX E
EVALUATION TASKS AND ADDITIONAL RESULTS

We evaluate on ten contact-rich tasks in three classes. *Binary tasks* (Tasks 1–7) require a sequence of grasps and in-hand adjustments and are scored 1 on success and 0 otherwise. *Continuous tasks* (Tasks 8 and 9) require repeated contacts with the right amount of force: the policy plucks each of ten indicated harp strings or strikes each of ten indicated xylophone bars, one note per position, and each rollout is scored by the number of these ten notes that sound cleanly, so partial progress is credited. A note counts only if it sounds cleanly: a pluck or strike that is too light produces no sound, and one that is too hard snaps or buzzes the string or bounces the mallet into a double hit. The target is not a melody but a clean sound at each indicated position. The *volume-adjustment* task (Task 10) requires fine dial manipulation; we report the number of trials (of ten) in which the dial is engaged and, over those trials, the mean absolute error of the final reading from the 400 μL target, since a never-engaged trial is a failure with no reading. Every task is evaluated over ten rollouts with object placement randomized per rollout (Figure 4), and all methods share the same scenes and objects. Each task figure below is followed by the task's text instruction, of which the prompt used in fine-tuning is a one-line abbreviation (e.g., "Play harp."), and by the behavior required for success; Figure 10 shows the full input modalities of the policy on one task: the text instruction, the three camera views, and the fingertip wrenches at one instant of a rollout. Figures 11 to 20 show key stages of one rollout per task (five frames sampled uniformly from the head camera, an external camera for the harp; for Task 7 the frames come from a teleoperated demonstration).

a) Failure modes: The failures we observe in the evaluation rollouts fall into three groups, all rooted in contact rather than in gross motion. *Lost contact without feedback.* In shoelace tying the critical moment is the first pinch: if the fingertips close beside the lace instead of on it, nothing in the images reveals the miss, and a policy without force feedback continues the crossing and looping motion open-loop on an empty hand, so every later step fails. Contact information is what allows a re-grasp here; the variant trained without force prediction never ties a bow (0%), whereas the full model recovers in some rollouts. *Force precision.* Pipetting, cap opening, and card shuffling fail at the level of how hard the fingers press rather than where they go. Pressing the plunger too hard past the first stop aspirates the wrong volume or air, while pressing too lightly leaves the liquid in the tip; the cap tasks fail when the pinch does not transmit enough torque to the cap; and in card shuffling too much thumb force bends the halves until cards fly out, while too little lets the half-deck slip out of the grip before the riffle. The harp is the extreme case: a pluck that is too forceful snaps or detunes a string, and one that is too light produces no note, so the score depends almost entirely on regulating fingertip force on a compliant object. *Dexterity under contact.* In screwdriver use the tip cams out of the screw head when the axial force drops during a turn, and the Wuji Hand slips in the left hand when its grip does not resist the torque; in multi-object grasping the first object is dropped while the fingers reconfigure for the second, and in pipette-volume adjustment the dial is never engaged in half of the trials; once engaged, four of five readings end within 4 μL of the target, so the failure is in committing to the contact rather than in regulating it. These groups separate the ablations: removing force prediction produces the open-loop and force-precision failures, while removing track prediction mostly produces misplaced but force-appropriate motions.

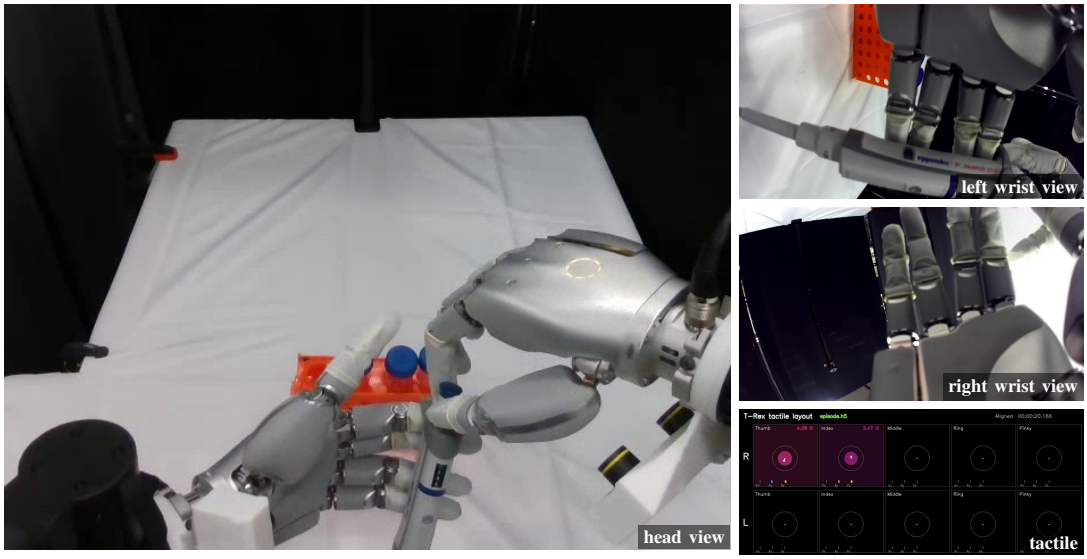


Fig. 10: **Full input modalities of FACT on Task 10: Adjust Pipette Volume.** *Text Instruction:* "A pipette set to 420 μL is on the table. Hold the pipette body with your left hand and turn the volume dial with your right thumb and index finger until the display reads 400 μL." Shown at one instant of an autonomous rollout while the dial is being turned: the head-camera view, the left and right ZED Mini wrist views, and the ten fingertip wrenches (rows R and L, one panel per finger; the dot gives the tangential force and the color the normal force), where the right thumb and index contacts on the dial appear while the left hand holds the pipette body. Together with the robot state, these are all the inputs the policy receives.

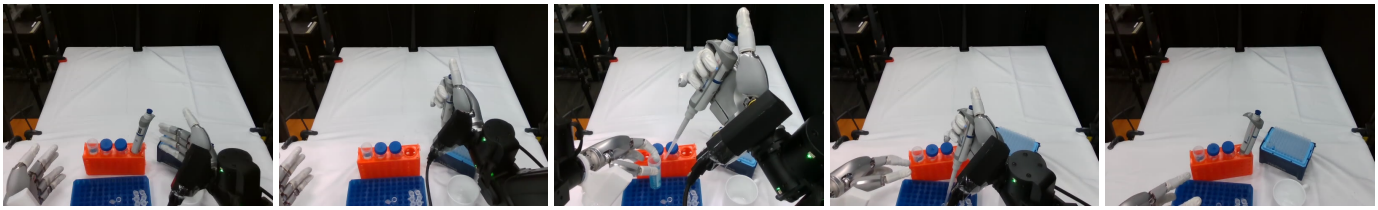


Fig. 11: Task 1: Lab Pipetting.

Task 1: Lab Pipetting. *Text Instruction:* "An orange rack holding a large tube and a small tube stands on the table with a pipette beside it; a box of pipette tips and a beaker sit in front of the rack. Pick up the pipette with your right hand and press it into a tip. Pick up the large tube from the rack with your left hand, draw liquid from it with the pipette, and place the tube back in the rack with your left hand. Then dispense the liquid into the small tube and eject the used tip into the beaker." The robot must pick up the pipette with the right hand in a working grip and press it into a tip, lift the large tube out of the rack with the left hand and hold it steady, lower the pipette into the tube and aspirate by pressing and releasing the plunger with the right index finger, return the tube to the rack with the left hand, dispense into the small tube by pressing the plunger again, and eject the tip into the beaker. A rollout succeeds only if liquid reaches the small tube and the tip is ejected without dropping the pipette or the tube.

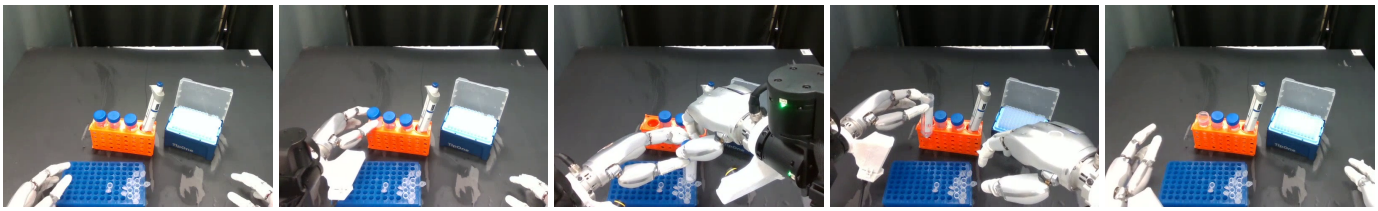


Fig. 12: Task 2: Open Test-Tube Cap.

Task 2: Open Test-Tube Cap. *Text Instruction:* "An orange rack holding capped test tubes stands on the table in front of you. Pick up one tube with your left hand, unscrew its cap with your right hand, and place the tube back in the rack with your left hand." The robot must lift a capped tube from the rack with the left hand and hold it upright, grasp the cap with the right thumb and index finger and unscrew it while the left hand holds the tube body against the twist, and return the tube to the rack with the left hand. A rollout succeeds only if the cap is fully separated from the tube and the tube stands in the rack at the end.

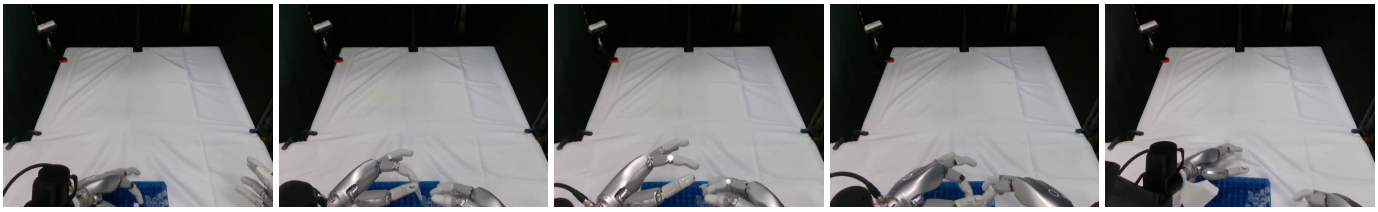


Fig. 13: Task 3: Open Microcentrifuge Flip Cap.

Task 3: Open Microcentrifuge Flip Cap. *Text Instruction:* "A blue rack holding microcentrifuge tubes sits at the front edge of the table. Pick up one tube with your left hand, pass it to your right hand, and flip its hinged cap open with your left thumb; then return the tube to the rack." The robot must lift a tube from the rack with the left hand, hand it over to the right hand, which holds the tube body, hook the left thumb under the lip of the cap and flip it open, and place the tube back in the rack. A rollout succeeds only if the cap is fully open and the tube stands upright in the rack.

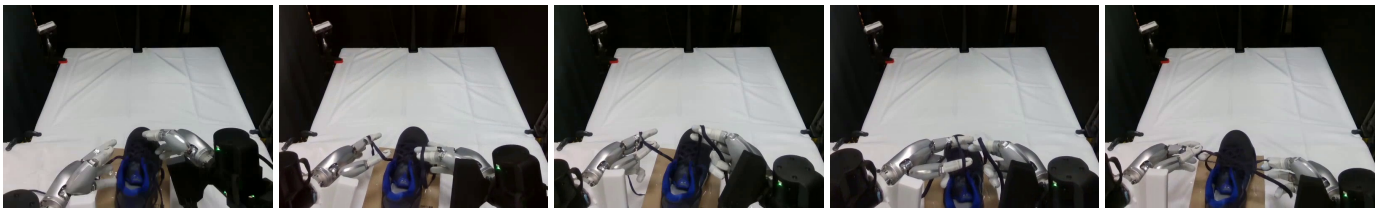


Fig. 14: Task 4: Tie Shoelaces.

Task 4: Tie Shoelaces. *Text Instruction:* "A shoe with its laces untied sits on a board in front of you. Using both hands, pick up the two laces, cross and tighten them, form a loop with one lace, wrap the other lace around it, pull it through to make a second loop, and tighten the bow." The robot must pinch the two laces with both hands, cross and tighten them, form a loop with one lace, wrap the other lace around it, push it through, and pull both loops tight. A rollout succeeds only if a two-loop bow remains tied after the hands release the laces.



Fig. 15: Task 5: Shuffle Cards.

Task 5: Shuffle Cards. *Text Instruction:* "Two halves of a deck of playing cards lie face down on a board in front of you. Pick up one half in each hand, bend the halves with your thumbs and riffle them together, then push the deck square." The robot must lift one half of the deck in each hand, bend both halves with the thumbs, release the cards so that they interleave, and push the deck square. A rollout succeeds only if the two halves merge into a single deck with no card dropped.

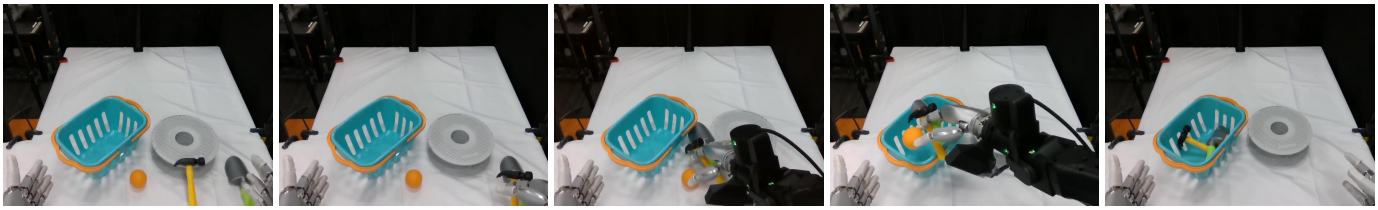


Fig. 16: Task 6: Multi-Object Grasp.

Task 6: Multi-Object Grasp. *Text Instruction:* "A basket, a plate, a toy shovel, a toy hammer, and a ping-pong ball are on the table. With your right hand, pick up the toy shovel, then the toy hammer, then the ping-pong ball without releasing the objects already in hand, and place them into the basket." The robot must pick up the toy shovel with the right hand, then grasp the toy hammer and the ping-pong ball with the same hand while keeping the objects already held, and place them into the basket. A rollout succeeds only if all three objects are held in the hand at the same time and end up in the basket.

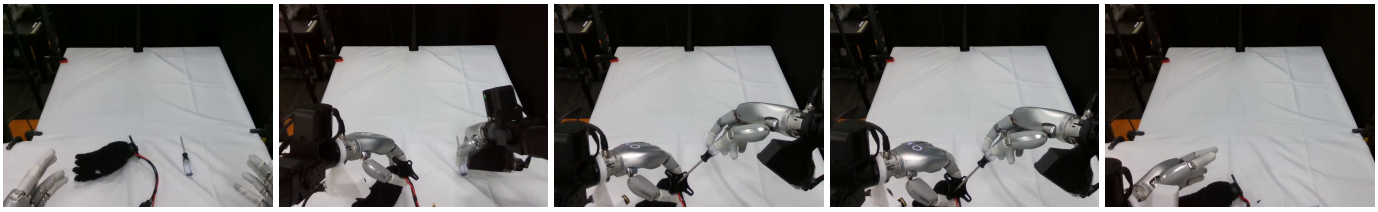


Fig. 17: Task 7: Use Screwdriver.

Task 7: Use Screwdriver. *Text Instruction:* "A second-generation Wuji Hand lies on the left of the table and a screwdriver on the right. Pick up the Wuji Hand with your left hand and hold it up with its base facing you, pick up the screwdriver with your right hand, seat its tip in the screw at the center of the base, and turn it counterclockwise until the screw comes out; then set the screwdriver down and place the hand back on the table." The robot must lift the Wuji Hand with the left hand and hold it steady in the air with the base exposed, grasp the screwdriver shaft with the right hand, bring the tip to the central screw of the base and seat it with light axial pressure, turn the screwdriver counterclockwise with repeated regrasps while the left hand resists the torque until the screw is free, and finally set the screwdriver down and return the hand to the table. A rollout succeeds only if the central screw is removed from the base and both objects are back on the table.

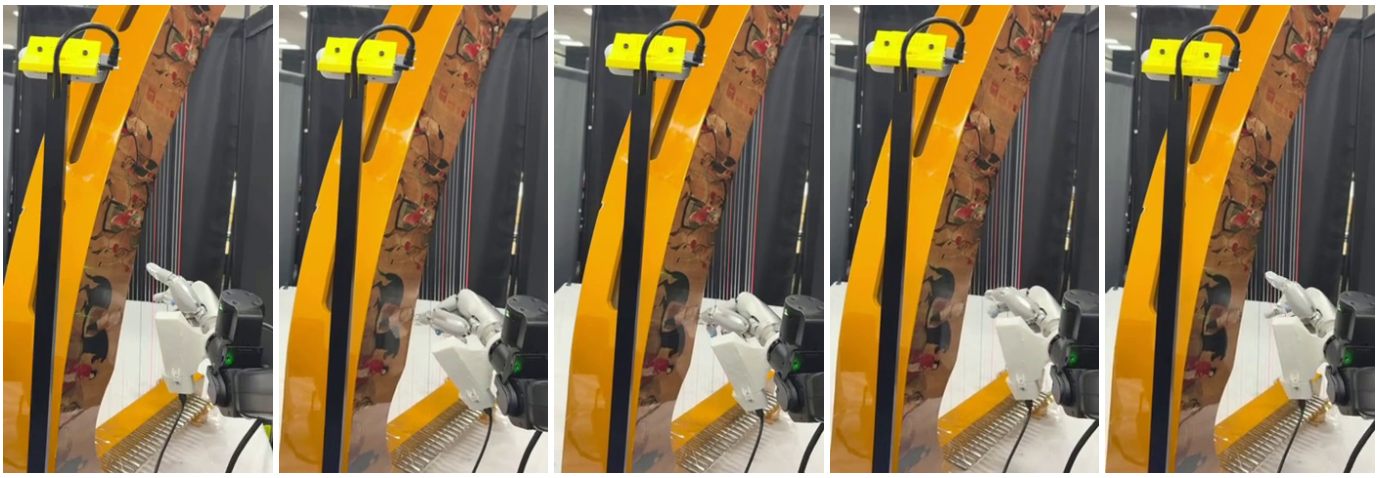


Fig. 18: Task 8: Play Harp.

Task 8: Play Harp. *Text Instruction:* "A harp stands in front of you. With your right hand, hook a fingertip behind each of the ten indicated strings in turn and pluck it so that it sounds." The robot must bring the right hand to the strings and, for each of the ten indicated strings, hook a fingertip behind it and release it with enough force to make the string ring cleanly, neither muting it nor snapping it. The score is the number of the ten notes that sound cleanly.

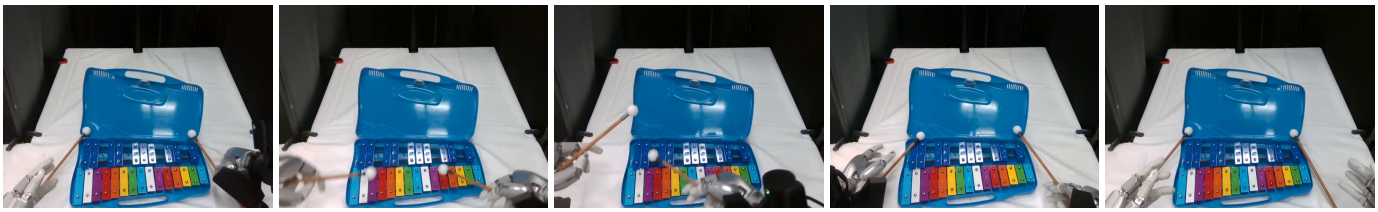


Fig. 19: Task 9: Play Xylophone.

Task 9: Play Xylophone. *Text Instruction:* "A xylophone lies open on the table with a mallet beside it. Pick up the mallet with your right hand and strike each of the ten indicated bars in turn so that it rings." The robot must pick up the mallet with the right hand, hold it in a stable grip, and strike each of the ten indicated bars with enough force to ring cleanly, neither a muted tap nor a bounced double hit. The score is the number of the ten notes that sound cleanly.

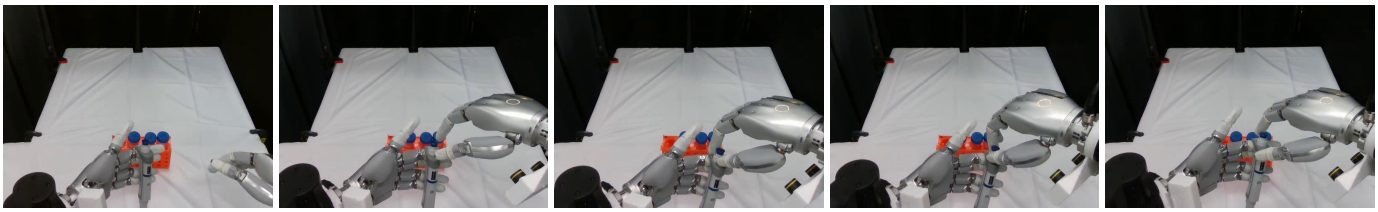


Fig. 20: Task 10: Adjust Pipette Volume.

Task 10: Adjust Pipette Volume. *Text Instruction:* "A pipette set to 420 μL is on the table. Hold the pipette body with your left hand and turn the volume dial with your right thumb and index finger until the display reads 400 μL." The robot must hold the pipette body with the left hand, roll the volume dial with the right thumb and index fingertip in small increments from 420 toward 400 μL, and stop when the display reads 400. A trial counts as engaged if the dial is turned; we report the engaged trials of ten and, over them, the mean absolute error of the final reading from 400 μL.