Robotics: Science and Systems 2026
Sydney, Australia, July 13–17, 2026

TactAlign: Human-to-Robot Policy Transfer via Tactile Alignment

Youngsun Wi[1], Jessica Yin[2], Elvis Xiang[1], Akash Sharma[3], Jitendra Malik[4], Mustafa Mukadam[5], Nima Fazeli[1,*], Tess Hellebrekers[6,*]
[1]University of Michigan [2]Nvidia [3]Carnegie Mellon University [4]UC Berkeley [5]University of Washington [6]Microsoft Research
[†]This work was partially done during Youngsun Wi's Meta FAIR internship. [*]Equal advising.

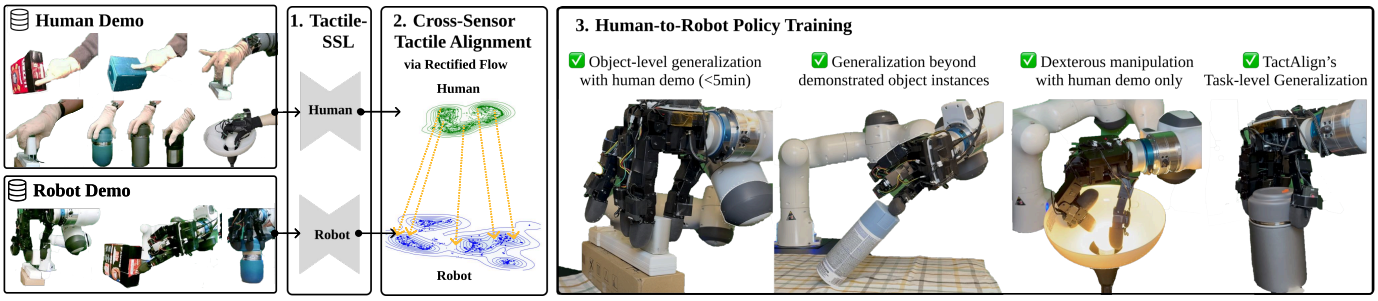


Fig. 1: We propose TactAlign, a cross-sensor tactile alignment method for cross-embodiment human-to-robot policy transfer. Given unpaired human (tactile glove) and robot demonstrations, TactAlign uses a rectified flow to map glove tactile features into the robot tactile space. This alignment enables effective tactile policy co-training on pivoting, insertion, and lid closing tasks. With only a few minutes of human demonstrations, the resulting policies generalize to unseen objects instances. Importantly, the same learned alignment can be reused to train policies on unseen tasks. We also demonstrate zero-shot human-to-robot dexterous manipulation on a light bulb screwing task.

***Abstract*—Human demonstrations collected by wearable devices (e.g., tactile gloves) provide fast and dexterous supervision for policy learning, and are guided by rich, natural tactile feedback. However, a key challenge is how to transfer human-collected tactile signals to robots despite the differences in sensing modalities and embodiment. Existing human-to-robot (H2R) approaches that incorporate touch often assume identical tactile sensors, require paired data, and involve little to no embodiment gap between human demonstrator and the robots, limiting scalability and generality. We propose TactAlign, a cross-embodiment tactile alignment method that transfers human-collected tactile signals to a robot with different embodiment. TactAlign transforms human and robot tactile observations into a shared latent representation using a rectified flow, without paired datasets, manual labels, or privileged information. Our method enables low-cost latent transport guided by hand-object interaction-derived pseudo-pairs. We demonstrate that TactAlign improves H2R policy transfer across multiple contact-rich tasks (pivoting, insertion, lid closing), generalizes to unseen objects and tasks with human data (≤ 5 minutes), and enables zero-shot H2R transfer on a highly dexterous tasks (light bulb screwing).**
WEBSITE: `yswi.github.io/tactalign/`

I. INTRODUCTION

As scaling data becomes increasingly important in robot learning, human demonstrations have emerged as a compelling data source: human trajectories can be collected 2-3x faster than robot teleoperation data [28], while also being inherently dexterous and multi-modal. Critically, our manipulation skills are guided by our rich, multi-sensory tactile feedback. However, most existing human-to-robot (H2R) approaches omit tactile feedback entirely and instead focus on transferring more readily available observations such as egocentric vision or state–action pairs in configuration space. As a result, tactile feedback, despite its central role in dexterous and contact-rich manipulation, has remained largely underexplored in H2R learning. This raises a key question: *how can human touch, collected through through wearable tactile devices, be effectively represented and transferred to robots?*

Recent work has begun to address this question by incorporating tactile sensing into human demonstrations for robot policy learning [45, 47, 1, 43, 9, 48, 40]. While effective, many of these approaches assume identical tactile sensors or little to no embodiment gap, which simplifies transfer but limits applicability across diverse robot hands. A concurrent work, UniTacHand [48], addresses cross-sensor tactile transfer, but relies on strict spatiotemporal correspondence between human and robot throughout task demonstrations. This strict pairing can be prohibitively difficult to maintain during contact-rich interactions involving sliding contact or dynamic object motion necessary for general manipulation. In contrast, our proposed method enables tactile transfer from unpaired datasets of the same task without requiring such pairing assumptions.

Beyond H2R settings, cross-sensor tactile transfer has also

been studied. However, existing methods primarily focus on static contact scenarios [30, 29, 11, 10] and coarse semantic-level alignment objectives [10, 51], leaving their effectiveness for continuous tactile reasoning during dynamic interactions such as sliding contact or object motion unclear. Moreover, these approaches often rely on paired supervision or labels, limiting scalability across heterogeneous sensors and robots.

We present TactAlign, a method to transfer human tactile observations collected with a wearable tactile device (e.g., a tactile glove [45]) to robots with heterogeneous tactile sensors. The objective of this cross-sensor tactile transfer is to enable effective cross-embodiment human-to-robot policy transfer. At a high level, TactAlign performs tactile alignment in two stages. First, it pretrains human and robot tactile encoders independently using self-supervised learning to obtain modality-specific latent representations. Second, it learns a cross-sensor alignment via rectified flow using pseudo-pairs derived from hand-object interactions, without requiring explicitly paired datasets. Rectified flow is well suited to learning mappings under noisy pseudo-pairs [23] arising from the non-unique relationship between hand-object motion and tactile observations. Together, these design choices simplify data requirements, allowing human demonstrations to be effectively leveraged for robot learning despite heterogeneous embodiment.

The core contributions of our work are:

- We propose TactAlign, a method for aligning cross-sensor tactile data from *unpaired* demonstrations of the same task. TactAlign leverages rectified flow with noisy pseudo-pairs to learn a latent mapping that enables H2R policy transfer between humans and robots equipped with heterogeneous tactile sensors.
- We show TactAlign improves H2R co-training success by $+59\%$ (vs. no tactile) and $+51\%$ (vs. no alignment), and generalizes to human-only objects ($+59\%$, vs. robot-only) and unseen objects ($+54\%$, vs. robot-only) using $\leq$5 minutes of human data across contact-rich tasks: pivoting, insertion, lid closing.
- Finally, we show that TactAlign enables zero-shot dexterous policy transfer from human data for light-bulb screwing, achieving a $+100\%$ improvement over policies trained without tactile input or alignment.

II. Related Works

A. Human-to-Robot Transfer

Prior work has shown leveraging fast and diverse human demonstrations enables scalable and generalizable robot policies. Most existing human-to-robot transfer methods operate in visual [40, 19, 28, 17, 4, 3] or kinematic spaces [38, 16, 22, 6, 12, 21]. More recently, a small number of works have begun incorporating tactile sensing into this paradigm, transferring human tactile signals to robots either via simplified grippers [47, 1] or through wearable systems tailored to specific robot hands [43, 9, 45]. These methods highlight the value of human tactile data by enabling demonstrators to feel and react to touch and transferring human tactile measurements to robot policy learning, but they assume humans and robots share the same tactile sensors. A concurrent work, UniTacHand [48], addresses cross-sensor tactile transfer, but relies on spatially and temporally strictly paired human-robot data. In contrast, TactAlign enables tactile transfer from unpaired datasets of the same task without requiring such strict correspondences.

B. Wearable Devices for Data-Collection

Recent works have explored wearable devices for more intuitive demonstration capture, particularly for dexterous robot hands. Compared to teleoperation, wearables allow the human demonstrator to interact directly with real objects with natural dexterity and haptic feedback, rather than controlling a different robot embodiment through intermediate devices and kinematic mismatches. However, current wearable devices necessitate a trade-off between the demonstrator's degrees of freedom (DOF) and tactile signal richness. While handheld and fingertip-based approaches [8, 47, 1] achieve high-fidelity 3D force sensing, they restrict the user to low-DOF parallel-jaw grasps. To support multi-fingered manipulation, exoskeleton devices [43, 9, 49] mechanically constrain human movement to match robot linkages, improving retargeting accuracy but sacrificing the dexterity of the human hand. Flexible gloves preserve the full DOFs of human dexterity but can be limited to kinematics-only data [38, 46] or normal forces only [24, 37]. In this work, we use the OSMO tactile glove [45], which combines the high dexterity of flexible gloves with rich shear and normal force sensing.

C. Cross-Sensor Tactile Alignment

A central challenge in tactile learning is handling heterogeneous sensor modalities while enabling effective knowledge transfer across them. Some prior works address this by learning shared representations across vision-based tactile sensors [13], but these approaches do not provide explicit one-to-one alignment or support direct cross-modal transfer. Other methods pursue explicit transfer through paired supervision or shared intermediate representations [30, 11, 10, 50, 29, 7]. However, these approaches either emphasize geometric aspects of touch, which limits their ability for dexterous manipulation that involves shear or sliding contact [29, 30, 11], focus on coarse categorical alignment [10], or rely on paired data collected using hand-designed 3D-printed objects [7, 29]. In contrast, our approach enables dense tactile alignment from natural task demonstrations, including sliding and dynamic motion, without requiring explicit labels or strictly paired data.

III. Methodology

A. Problem Statement

Our goal is to learn a latent-space mapping that transfers human tactile observations to robot tactile observations. We assume access to two offline datasets of task demonstration trajectories: one collected by a human wearing a tactile glove, denoted by H, and another collected from a dexterous robot with heterogeneous fingertip tactile sensors, denoted by R. The robot dataset is smaller in scale due to the

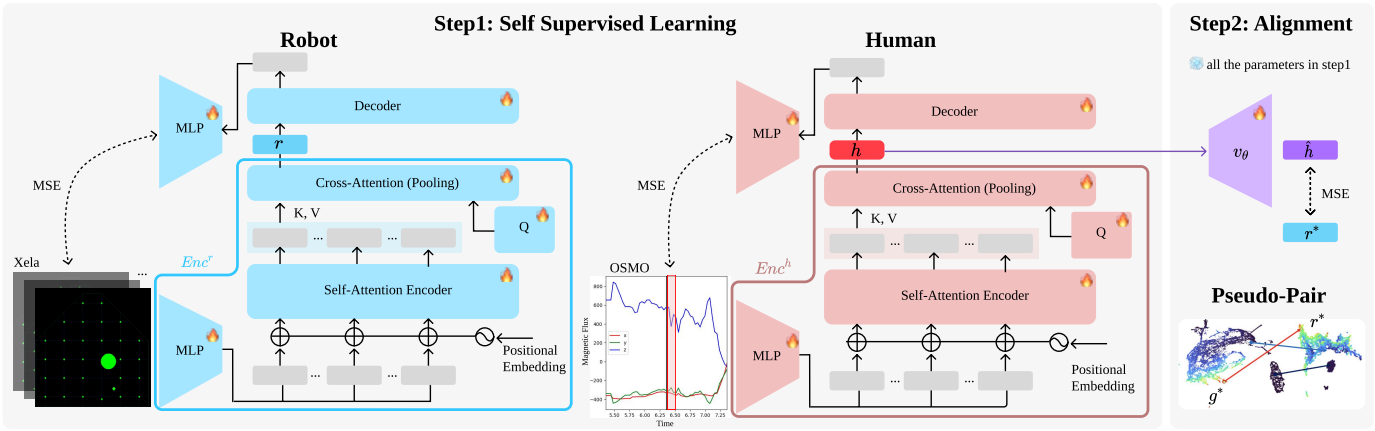


Fig. 2: **Tactile Alignment Overview.** Our method consists of two stages: self-supervised representation learning and cross-embodiment alignment via pseudo-pairs. We use a learnable length-1 query between the encoder and decoder to produce a fixed-dimensional latent representation via cross-attention pooling. A learnable length 1 query is implemented between the encoder and decoder to output a fixed-dimensional latent representations after the cross-attention module. In step2, we aggregate the learned latents from both domains to construct pseudo-pairs (h^*, r^*), and learn a velocity field v_θ that transports the glove latent distribution to the robot latent distribution.

higher cost of robot data collection. The datasets consist of demonstration trajectories, $H = \{\mathcal{T}_1^h, \ldots, \mathcal{T}_N^h\}$ and $R = \{\mathcal{T}_1^r, \ldots, \mathcal{T}_M^r\}$. Each human trajectory is represented as $\mathcal{T}^h = \{(F_1^h, P_1^h, w_1^h), \ldots, (F_l^h, P_l^h, w_l^h)\}$, while each robot trajectory is $\mathcal{T}^r = \{(F_1^r, P_1^r, w_1^r), \ldots, (F_l^r, P_l^r, w_l^r)\}$, where the length l varies across trajectories and t denotes the time step. At each time step t, tactile observations and poses from all K fingertips are represented as $F_t = (f_{t,1}, \ldots, f_{t,K})$, $P_t = (p_{t,1}, \ldots, p_{t,K})$, where $f_{t,k}$ and $p_{t,k}$ denote the tactile observation and pose of fingertip k, respectively, and w_t denotes the wrist pose. Superscripts h and r indicate human and robot.

We denote latent tactile mapping as $g : T^h \to T^r$, where $T^h, T^r \in \mathbb{R}^d$ are latent spaces derived from the human dataset H and the robot dataset R, respectively. We assume that a subset of trajectories in H and R, denoted by $A^h \subset H$ and $A^r \subset R$, correspond to the same task and share start and end object states. For these trajectories, we assume access to object pose estimates extracted from images, which are used only to establish initial correspondences. We represent these subsets as $A^h = \{(F_1^h, P_1^h, w_1^h, o_1^h), \ldots, (F_l^h, P_l^h, w_l^h, o_l^h)\}$, $A^r = \{(F_1^r, P_1^r, w_1^r, o_1^r), \ldots, (F_l^r, P_l^r, w_l^r, o_l^r)\}$, where o_t denotes the object pose at time t.

B. Tactile Self-supervised Learning

We represent per-fingertip tactile observations from two embodiments for the human glove and the robot. We use f_i^g, p_i^g and f_j^r, p_j^r to denote the tactile observation and pose of a single fingertip at independent time indices i and j for the human and robot trajectories, respectively. Here, the human tactile observation has size $w^h \times n^h \times d^h$, while the robot tactile signal is $w^r \times n^r \times d^r$, where w^h and w^r are a 0.1 second time window as in [34], n^h and n^r correspond to the spatial resolution, d^h and d^r are dimensions for the glove and robot, respectively. To accommodate the heterogeneous sensing modalities, we learn unique encoders and decoders for the human and robot tactile signals in a self-supervised manner, using a mean squared error (MSE) reconstruction loss to preserve modality-specific structure. Our architecture (Fig. 2 left) is based on JEPA [2] with the decoder adapted from the online probe module in [13, 34]. As a result, we get pretrained encoders for human and robot observations:

$$Enc^h(f_i^h) = h_i \in \mathbb{R}^d \quad \text{and} \quad Enc^r(f_i^r) = r_i \in \mathbb{R}^d, \tag{1}$$

where $\mathbb{R}^d$ is the shared latent tactile feature dimension. Since the two tactile signals differ in dimensionality, we implemented cross attention based pooling module at the end of each encoder for tactile features with consistent length [32].

C. Pseudo-Pair Extraction from Demonstrations

We extract human hand and object pose from each trajectory in a single pass, without requiring privileged information such as 3D object models (Appendix B-A). Then, we construct initial cross-domain correspondences from transitions:

$$O_i^h = (p_i^h, o_i^h, p_{i+1}^h, o_{i+1}^h), \quad O_j^r = (p_j^r, o_j^r, p_{j+1}^r, o_{j+1}^r).$$

We define the similarity metric between transitions as

$$\begin{aligned} S(O_i^h, O_j^r) = &\|p_i^h - p_j^r\| + \|o_i^h - o_j^r\| \\ &+ \lambda \left\|\hat{\Delta} p_i^h - \hat{\Delta} p_j^r\right\| + \lambda \left\|\hat{\Delta} o_i^h - \hat{\Delta} o_j^r\right\|. \end{aligned} \tag{2}$$

In this section, we operate in a normalized pose space, where positions and orientations are rescaled using task-level statistics; we reuse p and o to denote the normalized values (Appendix C-A). The scalar λ balances pose and velocity terms, and a single value is used across all (Appendix C-B).

Using this metric, we create a set of latent tactile pseudo pairs $P = \{(h_i^*, r_j^*) \mid S(O_i^h, O_j^r) < \delta\}$, where δ is a global similarity threshold. We construct the pseudo pairs from A^h and A^r between human-robot demonstrations from the same task, object, reset state and goal state. These pseudo pairs are inherently noisy and thus serve only as an initial alignment guide. We further refine P using binary contact

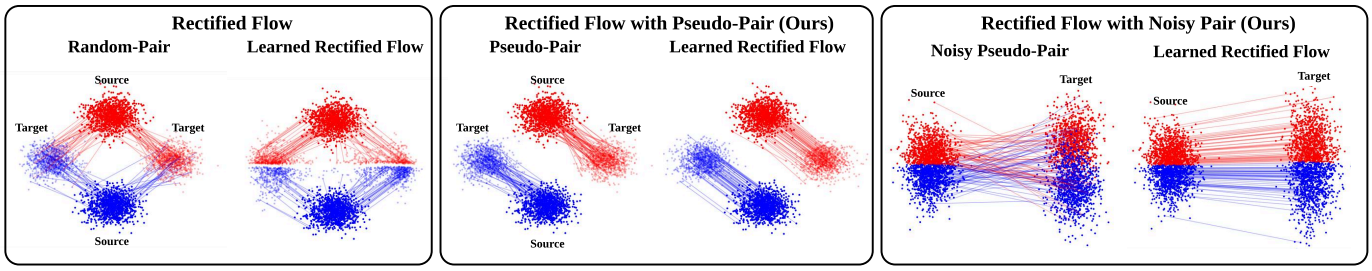


Fig. 3: Red and blue indicate two subsets of the source distribution. The left side of each of the three panels shows the provided training pairs (lines), with colors preserved at transparency $\alpha = 0.2$ for the target samples associated with each pair. The right side of each panel shows the transport from source to target under the learned flow. **First:** Standard rectified flow [23] learns a low-cost transport between two distributions by training on random pairs. **Second:** We propose incorporating pseudo-pairs into rectified flow to guide the velocity field toward desired correspondences between the source and target distributions. **Third:** Despite noise in the pseudo-pairs, the learned rectified flow remains robust and converges to an efficient transport map between the two distributions.

filtering. Specifically, if $\|f_i^h\| < \delta_h$ or $\|f_j^r\| < \delta_r$, the corresponding observations O_i^h and O_j^r are classified as non-contact; otherwise, they are treated as contact. We retain only pseudo pairs that map contact-to-contact and non-contact-to-non-contact states. This simple filtering is effective because contact transitions are often subtle in configuration space but are readily distinguishable in raw tactile observations.

D. Tactile Alignment via Rectified Flow

We formulate the cross-embodiment tactile alignment problem as a rectified flow [23] problem over a conditional distribution $p(x \mid t, z)$. The variable x denotes a latent tactile state evolving over normalized time $t \in [0, 1]$, while the conditioning variable $z = (h_i^*, r_j^*)$ is obtained from pseudo pairs constructed offline across different tasks. Unlike the original rectified flow [23], which relies on random pairings, we guide the flow using pseudo-pairs extracted from offline hand-object interactions, providing coarse initial correspondences (Fig. 3).

Our goal is to learn a velocity field v_θ that transports human tactile features to the robot's (Fig. 2 right). Specifically, for each pseudo pair $(h_i^*, r_j^*) \in P$, we define an interpolated latent state $x_t = t \cdot h_i^* + (1-t) r_j^*$ and constant velocity $h_i^* - r_j^*$ at x_t. The velocity field v_θ is trained by solving a least-squares regression problem over $t \in [0, 1]$:

$$\min_{v_\theta} \sum_{(h_i^*, r_j^*) \in P} \int_0^1 \left\| (h_i^* - r_j^*) - v_\theta(x_t, t) \right\|^2 dt. \tag{3}$$

This process naturally performs latent "rewiring", effectively handling crossings and transport cost reductions while learned from noisy pairs as in Fig. 3. During inference time, we simulate the resulting ODE, $dx_t = v_\theta(x_t, t)$, and transform human tactile feature from h_i to $\hat{h}_i$ as

$$g_\theta(h_i) = \hat{h}_i = \int_0^1 v_\theta(x_t, t) dt \quad \text{with } x_0 = h_i. \tag{4}$$

In practice, we solve Eq. 4 via vanilla Euler method with constant step size [23]. Theoretically, the ODE can be solved either forward and backward as Eq. 3 is time-symmetric. Architecture and training details are in Appendix D-A.

E. Human to Robot Policy Learning

We learn a H2R policy, adapted from ACT [51], which is shared across both human and robot embodiments (Fig. 4). The policy is defined as $\pi_\phi(\{\hat{h}_{t,k}\}_{k=1}^K, P_t^h, w_t^h) = a_t^h$ and $\pi_\phi(\{r_{t,k}\}_{k=1}^K, P_t^r, w_t^r) = a_t^r$. Here, $\{\hat{h}_{t,k}\}_{k=1}^K$ and $\{r_{t,k}\}_{k=1}^K$ denote per-fingertip tactile latent features for the human and robot, respectively. Both outputs a_t^h, a_t^r consist of action chunks specifying desired fingertip locations and wrist orientation with respect to the robot base frame. For human demonstrations, the wrist orientation is adjusted by a constant offset equal to the difference between the average human and robot wrist orientations over the entire trajectory [16]. During execution, the policy runs at 10-30 Hz (Appendix E).

IV. Experiments and Results

A. Hardware

In our experiments, humans wear the OSMO glove [45], which provides three-axis magnetic tactile signals at the fingertips. Robot demonstrations are collected using Xela sensors

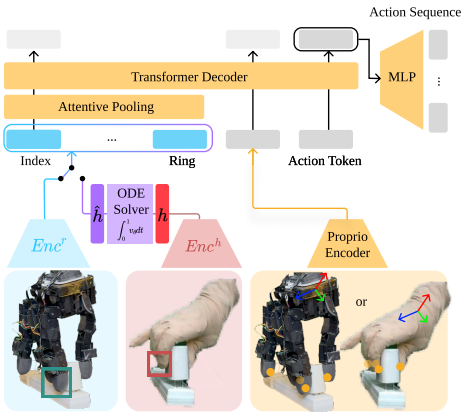


Fig. 4: **H2R Action Policy.** Given either human or robot inputs, the shared policy follows a color-coded structure, representing robot, human, and shared modules. Human glove latent features are passed into an ODE solver via a learned velocity field. The proprioceptive encoder takes fingertip locations in yellow dots and wrist orientation. Only the yellow modules are trained; all others are frozen.

mounted on the Allegro Hand fingertips. While both sensors are magnetic-based, they differ in sensing mechanisms, as OSMO uses a particle-based magnetic skin whereas Xela employs discrete magnet-based sensing, resulting in differences in signal scale and characteristics as well as spatial resolution (OSMO: 1×3; Xela: 30×3). We use a Franka Emika Panda arm and a RealSense D455 camera, and employ an ATI Gamma force–torque (F/T) sensor only in Sec. IV-F.

B. Dataset

We summarize the datasets used at each step below. Details on tactile signal post-processing are provided in Appendix B-B.

1) ***Tactile self-supervised learning****:* Both human and robot tactile encoders are trained using a combination of play data (≈10 minutes) and an in-domain tactile alignment dataset.

2) ***Tactile alignment via rectified flow****:* We use a total of 100 robot demonstrations via kinesthetic teaching and 200 human demonstrations collected from two contact-rich manipulation tasks: pivoting and insertion. Each task involves a single shared object, and we intentionally collect diverse human demonstrations. At this stage only, we focus on tasks in which the object pose changes relative to the hand or world during interaction.

3) ***Human-robot policy co-training****:* We chose three representative contact-rich tasks: pivoting, insertion, and lid-closing. For each task, we collect 140–160 human demonstrations, where 100 demonstrations (≈ 30 minutes) are from the same object seen by the robot ("seen-by-both" object), and 20 demonstrations are collected for each additional "human-only" object (≈ 5 minutes per object). For robot data, we collect 50 via kinesthetic using a single training object (≈ 60 minutes).

4) ***Dexterous policy learning with human-only****:* Here, we collect human data only, consisting of 20 demonstrations of light-bulb screwing. We use Manus glove [25] with OSMO tactile sensors [45] for robust hand pose estimation under visual occlusions from the lamp shade and light bulb. We record fingertip poses only, as the Manus glove does not provide wrist pose information. During data collection, human demonstrators screw a light bulb into a fixed lamp; the human wrist pose is randomized across demonstrations but remains static within each demonstration.

5) ***Force prediction evaluation****:* We collect a force-labeled dataset using an F/T sensor for both robot and human tactile observations (Appendix A-C). The dataset includes 1,472 robot force samples (24:1 train:test split) and 1,527 human force samples used only for evaluation. Robot force data are used exclusively for training, while human force measurements are reserved for testing (Appendix IV-F).

C. Learned Rectified Flow

We evaluate the learned human-to-robot rectified flow using UMAP projections [26] in Fig. 5 for both the pivoting and insertion tasks. After alignment, the human tactile features measured from the glove move toward the robot features and nearly overlap. Interestingly, we observe a consistent

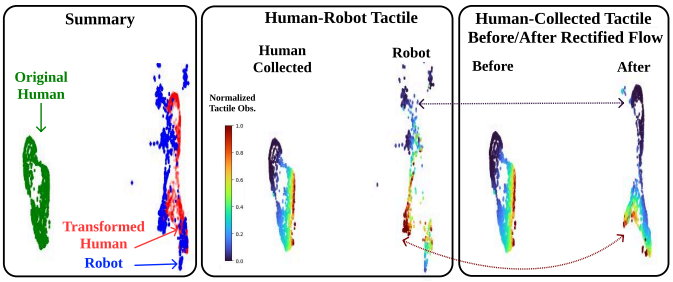


Fig. 5: **Tactile Features UMAP Projections. First:** Rectified flow maps the glove latent distribution to overlap with the robot distribution. **Second & Third:** Colors denote normalized raw tactile magnitude (0: no contact, 1: highest force/shear), computed separately for glove and robot data. As indicated by the arrows, the alignment exhibits a consistent cross-domain trend in contact force magnitudes, even though force is not used during training.

trend in normalized force magnitudes across domains: glove features associated with higher contact forces tend to map to robot features with similarly high normalized forces, and vice versa, despite force never being explicitly used during training. Quantitatively, the Earth Mover's Distance (EMD) [15], which measures distance between distributions while accounting for their geometry, between the human and robot tactile distributions decreases by 78% after alignment, from 0.091 to 0.020. Further force-centric qualitative analysis is in Sec. IV-F.

D. Human-Robot Policy Co-Training

We evaluate the effectiveness of TactAlign for H2R policy co-training on three representative contact-rich manipulation tasks in Tab. II. All three tasks require force reasoning and begin from a non-contact state, either between the fingertip and the object or between a randomly grasped object and the environment. Successful execution therefore depends on detecting contact onset and reasoning about contact throughout the task as in Fig. 6, 7, and 8. The pivoting and insertion tasks evaluate generalization to unseen objects within the same task, while the lid closing task additionally tests our alignment module's generalization to an unseen task class not used during training. We report the co-training results across all tasks in Tab. I. For all tasks, we roll-out 10 times per each object and compare four settings:

1) ***Robot-only baseline****:* The robot-only baseline in Tab. I evaluates a policy trained exclusively on robot-collected data using a seen-by-both object. This baseline isolates the contribution of human demonstrations to policy generalization. We observe that augmenting robot training with human demonstrations, which are substantially easier to collect and $\approx 4\times$ faster in our setting, markedly improves generalization to unseen objects. Without tactile modality, performance improves +10% on seen-by-both objects, +59.3% on human-only objects, and +54.4% on held-out objects.

2) ***Without Tactile baseline****:* The TactAlign w/o tactile baseline in Tab. I evaluates an H2R policy trained without tactile input, relying only on proprioception. This baseline probes the extent to which each task depends on tactile feedback and

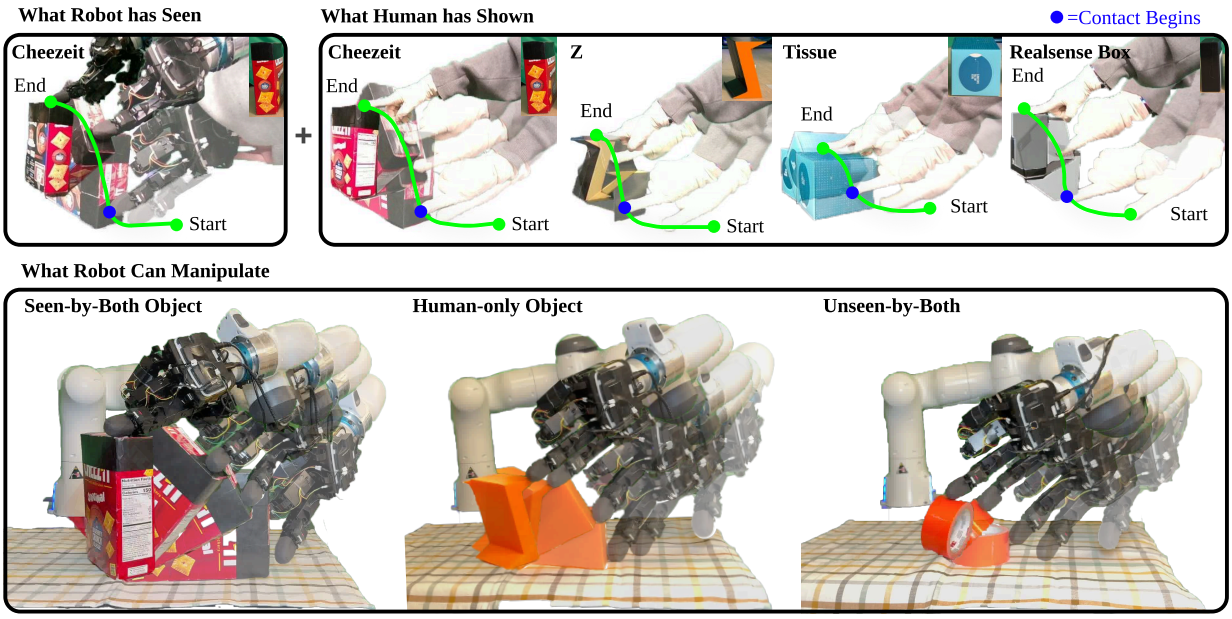


Fig. 6: **Pivoting Task.** The task begins in a non-contact state and transitions to pivoting upon contact detection via tactile feedback, with the goal of maintaining contact without dropping the object. **Top**: training demonstrations. **Bottom**: robot policy rollouts.

Task	**All**	**Seen for Alignment (v_θ, Seen-by-both only)**						**Unseen for Alignment (v_θ)**		
		Pivoting			**Insertion**			**Lid Closing**		
Method [%] ↑	Avg.↑	Seen-by-both	Human-only	Unseen-by-both	Seen-by-both	Human-only	Unseen-by-both	Seen-by-both	Human-only	Unseen-by-both
Robot Only	38	100	0	0	70	0	33	100	35	0
TactAlign w/o Tactile	21	0	0	0	50	35	43	20	0	40
TactAlign w/o Align	28	0	63	13	10	5	0	70	40	55
TactAlign	**79**	**100**	**83**	**60**	**100**	**65**	**67**	**100**	**65**	**70**

TABLE I: **H2R Co-training.** We evaluate human-to-robot transfer on tasks used for alignment (pivoting and insertion) as well as on an unseen task (lid closing). Performance is reported across three object categories: seen-by-both, human-only, and held-out objects. Incorporating human data enables zero-shot generalization to objects not observed during robot training and beyond those present in the human dataset. We compare against three baselines: robot-only training, TactAlign without tactile input, and TactAlign without tactile alignment.

highlights the role of touch in effective H2R learning. We observe that incorporating tactile input substantially improves performance across all three tasks, increasing success rates +59% on average. The largest performance gap appears in the pivoting task, where success improves by up to +100%, followed by lid-closing and insertion. These trends suggest that tasks involving more diverse hand-object interactions, as reflected in the action space, benefit more from tactile feedback, since reliance on proprioception alone provides insufficient information for successful task execution.

*3) **Without Alignment baseline**:* The TactAlign w/o align baseline in Tab. I evaluates an H2R policy trained with TactAlign but without tactile alignment, using raw tactile features h_i instead of aligned features $\hat{h}_i$. This baseline isolates the contribution of tactile alignment in H2R co-training. Removing tactile alignment results in a substantial degradation in performance, with an overall -51% drop in average success rate compared to the full method. Interestingly, non-aligned tactile features are often detrimental, leading to near-complete failure on seen-by-both objects for the pivoting and insertion tasks. For the lid-closing task, which is not used during alignment training, the performance gap between TactAlign and TactAlign w/o align is smaller (-23% drop). We hypothesize that lid closing admits a broader set of successful contact strategies and can benefit from coarse contact cues present in raw tactile signals, even without consistent cross-embodiment semantics. Together, these results suggest that without alignment, raw tactile features can introduce inconsistent cross-embodiment semantics that hinder policy learning, particularly for tasks requiring precise contact interactions.

*4) **TactAlign**:* The TactAlign results in Tab. I correspond to our full method. Across all three object categories, TactAlign demonstrates consistent performance on both tasks used for alignment (pivoting, insertion) and a task not used during alignment or encoder training (lid closing). Specifically, TactAlign achieves success rates of 76%, 72%, and 74% on the pivoting, insertion, and lid-closing tasks, respectively. It also attains perfect performance on seen-by-both objects (100%) and maintains strong generalization to human-only (71%) and held-out objects (65.5%), averaged across three tasks. These results highlight the benefit of tactile alignment combined with data diversity for robust cross-task and cross-object generalization.

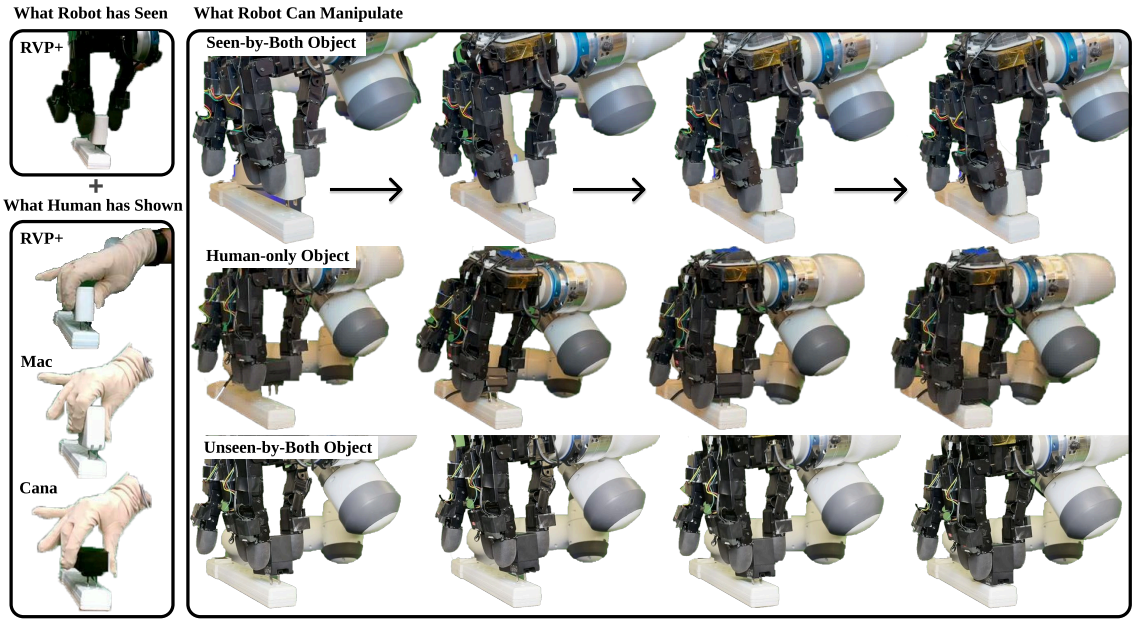


Fig. 7: **Insertion Task.** With randomized grasps, the policy leverages tactile feedback to perform search, alignment, and insertion of the adapter into the outlet. We show human demonstrations helps generalization to unseen adapters across variations in geometry, size, and mass.

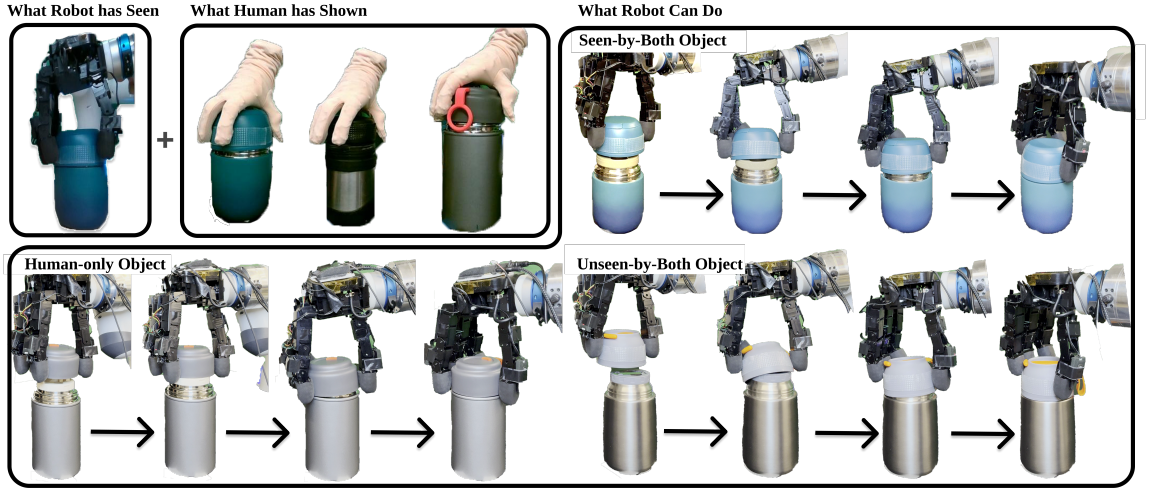


Fig. 8: **Lid Closing Task.** With randomized grasps, the policy uses touch to perform search, alignment, and closing between the lid and the bottle. We show human data improves generalization to unseen objects with varying lid and bottle geometries, sizes, and masses.

E. Dexterous Robot Policy Learning with Human Data Only

The light-bulb screwing task in Fig. 9 represents a dexterous and occlusion-heavy manipulation scenario, often due to the presence of lamp shades and small objects compared to robot fingers. As a result, teleoperation is non-trivial for this dexterous task [46], particularly in the absence of tactile feedback and reliable visual cues. In this task, we highlight two aspects of human-to-robot transfer: (1) the dexterity enabled by human demonstrations, in which demonstrators rely on tactile feedback to guide precise finger motions during screwing, and (2) zero-shot human-to-robot policy execution. As in Tab. III, our TactAlign policy achieves a 100% success rate. On average, screwing the light bulb until illumination takes approximately 61 seconds. In contrast, without tactile input, the success rate drops to 0%. The primary failure mode is the inability of the fingertips to establish stable contact with the light bulb, preventing task execution altogether. Without alignment, the success rate is also 0%, with failures primarily arising from jamming, from which the policy cannot recover, often leading to complete unscrewing of the light bulb. Compared to H2R co-training (Tab. I), the performance gap between the two baselines is largest, highlighting the importance of tactile alignment, especially when robot data are scarce.

F. H2R Force Estimation

We quantitatively evaluate H2R tactile alignment using a cross-sensor force prediction task, where neither tactile

	Pivoting		Seen-by-both	Human-only			Unseen-by-both		
	Object	Avg.↑	Cheezit	Z	Tissue	Realsense	Tape	Spray	CAN Box
Specs	Length [mm]	149	210	140	130	146	100	200	120
	Weight [g]	224	558	117	191	143	146	366	45
Methods↑	Robot Only	14	100	0	0	0	0	0	0
	Ours w/o Tactile	0	0	0	0	0	0	0	0
	Ours w/o Align	34	0	80	80	30	40	10	0
	Ours	**76**	**100**	**100**	**90**	**60**	**30**	**70**	**80**

	Insertion		Seen-by-both	Human-only		Unseen-by-both		
	Object	Avg.↑	RVP+	Mac	Cana	Belkin	TKDY	FNRSi
Specs	Height [mm]	64	66	80	50	66	50	70
	Weight [g]	114	83	165	90	80	88	176
Methods↑	Robot Only	28	70	0	0	50	10	40
	Ours w/o Tactile	42	50	20	50	40	50	40
	Ours w/o Align	12	0	10	0	10	50	0
	Ours	**72**	**100**	**60**	**70**	**60**	**50**	**90**

	Lid Closing		Seen-by-both	Human-only		Unseen-by-both	
	Object	Avg.↑	Haweek	Thermos	Cyxw	Energify	Haers
Specs	Height [mm]	115	95	100	140	130	110
	Width [mm]	95	100	85	95	93	100
Methods↑	Robot Only	34	100	70	0	0	0
	Ours w/o Tactile	20	20	0	0	40	40
	Ours w/o Align	52	70	70	10	60	50
	Ours	**74**	**100**	**90**	**40**	**70**	**70**

TABLE II: **H2R Co-training Results.** Success rates (%) averaged over 10 rollouts per object for pivoting, insertion, and lid closing. **Avg.** reports mean performance across all objects for each task. Each object is shown in Sec. A-E.

Light Bulb Screwing	Ours w/o Tactile	Ours w/o Align	Ours
Success rate↑[%]	0	0	100

TABLE III: **Light Bulb Screwing Result.** Success rates on the light bulb screwing task, evaluated over 10 rollouts.

observations nor force measurements are seen during encoder or alignment training. This evaluates whether the aligned latent space preserves physically meaningful information that transfers across heterogeneous tactile sensors.

*1) **Evaluation**:* We train a robot force decoder $D^r(r_j) \in \mathbb{R}^3$ that predicts contact forces from robot tactile features. D^r is trained on top of frozen TactAlign analogous to a linear probe as in [2] (Appendix D-E). We evaluate force prediction under three settings: *(i)* $H \rightarrow R$ *without* alignment, where human tactile features are directly decoded as $D^r(h_i)$; *(ii)* $H \rightarrow R$ with TactAlign, where the human tactile features are first mapped into the aligned latent space and decoded as $D^r(g_\theta(h_i))$; and *(iii)* $R \rightarrow R$, where the evaluation are performed on heldout robot data, representing a best-case upper bound. We evaluate five times and report the the mean and standard deviation of the per-axis ℓ_1 force prediction error.

*2) **Analysis**:* Fig. 10 reports the ℓ_1 force prediction error along each axis. In the $H \rightarrow R$ setting without alignment, force prediction errors are high across all axes, while TactAlign reduces the force prediction error by approximately 98%, 99%, and 93% along the F_x, F_y, and F_z axes, respectively, while also significantly reducing variance across runs. One reason for the large error without alignment is that the force decoder D^r, trained only on robot tactile features, must extrapolate across the substantial distribution gap. To contextualize, we compare against the robot-to-robot ($R \rightarrow R$) baseline, which represents a best-case upper bound where both training and evaluation are performed on robot tactile data. With alignment, $H \rightarrow R$ performance reaches within 2% of the $R \rightarrow R$ baseline on F_x and within 13% on F_y. A larger gap remains on F_z, where the aligned error is on average 0.7N higher than the $R \rightarrow R$ baseline. Nevertheless, alignment recovers a large portion of the performance gap, implying TactAlign's ability to transfer physically meaningful force information between heterogeneous tactile sensors.

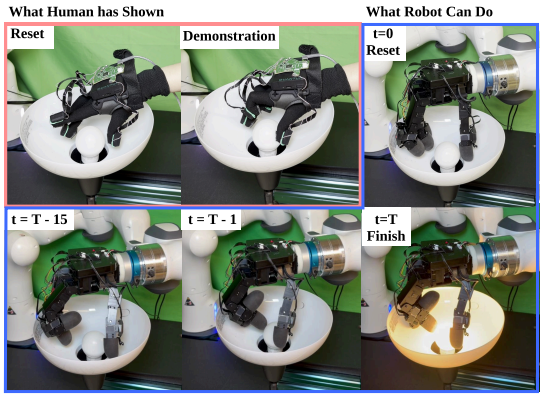


Fig. 9: **Light Bulb Screwing Task.** TactAlign learns a dexterous tactile policy using only human demonstrations, with zero robot data.

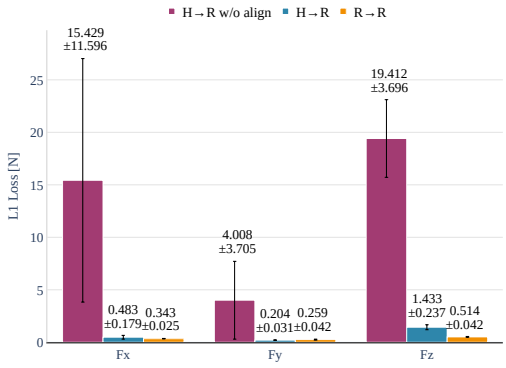


Fig. 10: ℓ_1 force prediction error (mean $\pm$ std) along each axis, averaged over five evaluations. $G \rightarrow R$ evaluates force prediction on the robot using human tactile signals, with (blue) and without alignment (red) alignment. With alignment reduces the force prediction error by 96.75% across the three force axes. $R \rightarrow R$ denotes the robot-to-robot baseline (yellow), representing a best-case scenario.

V. Limitation

While the formulation of TactAlign is sensor-agnostic, our evaluation is limited to a single glove-robot pairing due to hardware availability. Extending the approach to additional tactile modalities (e.g., vision-based tactile sensors), multi-hand settings, or full-palm sensing is left for future work. Moreover, tactile alignment alone does not address visual discrepancies between human and robot embodiments. Incorporating vision and other modalities into a unified multi-modal policy is also an important direction for future work.

ACKNOWLEDGMENTS

We thank Chan Hee Song, Mark van der Merwe, Tingfan Wu, Taosha Fan, and François Hogan for valuable project discussions. We are grateful to Mike Lambeta, James Lorenz, and Fan Yang for their support with hardware debugging, and to Woniks for their Allegro Hand hardware support.

REFERENCES

[1] Ademi Adeniji, Zhuoran Chen, Vincent Liu, Venkatesh Pattabiraman, Raunaq Bhirangi, Siddhant Haldar, Pieter Abbeel, and Lerrel Pinto. Feel the force: Contact-driven learning from humans. *arXiv preprint arXiv:2506.01944*, 2025.

[2] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[3] Homanga Bharadhwaj, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. Zero-shot robot manipulation from passive human videos. *arXiv preprint arXiv:2302.02011*, 2023.

[4] Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *Conference on Robot Learning*, 2025.

[5] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts, 2025. URL https://arxiv.org/abs/2511.16719.

[6] Mingfei Chen, Yifan Wang, Zhengqin Li, Homanga Bharadhwaj, Yujin Chen, Chuan Qin, Ziyi Kou, Yuan Tian, Eric Whitmire, Rajinder Sodhi, et al. Flowing from reasoning to motion: Learning 3d hand trajectory prediction from egocentric human interaction videos. *arXiv preprint arXiv:2512.16907*, 2025.

[7] Zhuo Chen, Ni Ou, Xuyang Zhang, Zhiyuan Wu, Yongqiang Zhao, Yupeng Wang, EmmanouilSpyrakos Papastavridis, Nathan Lepora, Lorenzo Jamone, Jiankang Deng, and Shan Luo. Training tactile sensors to learn force sensing from each other. *Nature Communications*, 2026.

[8] Hojung Choi, Yifan Hou, Chuer Pan, Seongheon Hong, Austin Patel, Xiaomeng Xu, Mark R Cutkosky, and Shuran Song. In-the-wild compliant manipulation with umi-ft. *arXiv preprint arXiv:2601.09988*, 2026.

[9] Hao-Shu Fang, Branden Romero, Yichen Xie, Arthur Hu, Bo-Ruei Huang, Juan Alvarez, Matthew Kim, Gabriel Margolis, Kavya Anbarasu, Masayoshi Tomizuka, et al. Dexop: A device for robotic transfer of dexterous human manipulation. *arXiv preprint arXiv:2509.04441*, 2025.

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

[11] Francesco Grella, Alessandro Albini, Giorgio Cannata, and Perla Maiolino. Touch-to-touch translation-learning the mapping between heterogeneous tactile sensing technologies. In *2025 IEEE 21st International Conference on Automation Science and Engineering (CASE)*, pages 1998–2004. IEEE, 2025.

[12] Irmak Guzey, Haozhi Qi, Julen Urain, Changhao Wang, Jessica Yin, Krishna Bodduluri, Mike Lambeta, Lerrel Pinto, Akshara Rai, Jitendra Malik, et al. Dexterity from smart lenses: Multi-fingered robot manipulation with in-the-wild human demonstrations. *arXiv preprint arXiv:2511.16661*, 2025.

[13] Carolina Higuera, Akash Sharma, Chaithanya Krishna Bodduluri, Taosha Fan, Patrick Lancaster, Mrinal Kalakrishnan, Michael Kaess, Byron Boots, Mike Lambeta, Tingfan Wu, and Mustafa Mukadam. Sparsh: Self-supervised touch representations for vision-based tactile sensing. In *8th Annual Conference on Robot Learning*, 2024.

[14] Humanoid Guide. Wuji hand. https://www.humanoid.guide/product/wuji-hand/. Accessed: 2026-02-03.

[15] Leonid V Kantorovich. Mathematical methods of organizing and planning production. *Management science*, 6 (4):366–422, 1960.

[16] Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 13226–13233. IEEE, 2025.

[17] Simar Kareer, Karl Pertsch, James Darpinian, Judy Hoffman, Danfei Xu, Sergey Levine, Chelsea Finn, and Suraj Nair. Emergence of human to robot transfer in vision-language-action models, 2025. URL https://arxiv.org/abs/2512.22414.

[18] Chung Min Kim*, Brent Yi*, Hongsuk Choi, Yi Ma, Ken Goldberg, and Angjoo Kanazawa. Pyroki: A modular toolkit for robot kinematic optimization. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025. URL https://arxiv.org/abs/2505.03728.

[19] Hanjung Kim, Jaehyun Kang, Hyolim Kang, Meedeum Cho, Seon Joo Kim, and Youngwoon Lee. Uniskill: Imitating human videos via cross-embodiment skill representations. *Conference on Robot Learning*, 2025.

[20] Taeyeop Lee, Bowen Wen, Minjun Kang, Gyuree Kang, In So Kweon, and Kuk-Jin Yoon. Any6D: Model-free 6d pose estimation of novel objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference (The IEEE/CVF Conference on Computer Vision and Pattern Recognition)*, 2025.

[21] Marion Lepert, Jiaying Fang, and Jeannette Bohg. Masquerade: Learning from in-the-wild human videos using data-editing. *arXiv preprint arXiv:2508.09976*, 2025.

[22] Vincent Liu, Ademi Adeniji, Haotian Zhan, Siddhant Haldar, Raunaq Bhirangi, Pieter Abbeel, and Lerrel Pinto. Egozero: Robot learning from smart glasses. *arXiv preprint arXiv:2505.20290*, 2025.

[23] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023.

[24] Yiyue Luo, Murphy Wonsick, Jessica Hodgins, and Brian Okorn. Tactile embeddings for multi-task learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[25] Manus Meta. Manus meta glove. https://www.manus-meta.com/, 2024. Accessed: Jan. 30, 2026.

[26] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software*, 3 (29):861, 2018.

[27] Rolandos Alexandros Potamias, Jinglei Zhang, Jiankang Deng, and Stefanos Zafeiriou. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12242–12254, 2025.

[28] Ryan Punamiya, Dhruv Patel, Patcharapong Aphiwetsa, Pranav Kuppili, Lawrence Y Zhu, Simar Kareer, Judy Hoffman, and Danfei Xu. Egobridge: Domain adaptation for generalizable imitation from egocentric human data. In *Human to Robot: Workshop on Sensorizing, Modeling, and Learning from Humans*, 2025.

[29] Samanta Rodriguez, Yiming Dou, Miquel Oller, Andrew Owens, and Nima Fazeli. Cross-sensor touch generation. *Conference on Robot Learning*, 2025.

[30] Samanta Rodriguez, Yiming Dou, William van den Bogert, Miquel Oller, Kevin So, Andrew Owens, and Nima Fazeli. Contrastive touch-to-touch pretraining. 2025.

[31] Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings third international conference on 3-D digital imaging and modeling*, pages 145–152. IEEE, 2001.

[32] Cicero dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. Attentive pooling networks. *arXiv preprint arXiv:1602.03609*, 2016.

[33] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964.

[34] Akash Sharma, Carolina Higuera, Chaithanya Krishna Bodduluri, Zixi Liu, Taosha Fan, Tess Hellebrekers, Mike Lambeta, Byron Boots, Michael Kaess, Tingfan Wu, et al. Self-supervised perception for tactile skin covered dexterous hands. *arXiv preprint arXiv:2505.11420*, 2025.

[35] Sharpa. Clone hand. https://clonerobotics.com/hand, . Accessed: 2026-02-03.

[36] Sharpa. Sharpawave dexterous robotic hand. https://www.sharpa.com/, . Accessed: 2026-02-03.

[37] Subramanian Sundaram, Petr Kellnhofer, Yunzhu Li, Jun-Yan Zhu, Antonio Torralba, and Wojciech Matusik. Learning the signatures of the human grasp using a scalable tactile glove. *Nature*, 2019.

[38] Tony Tao, Mohan Kumar Srirama, Jason Jingzhou Liu, Kenneth Shaw, and Deepak Pathak. Dexwild: Dexterous human interactions for in-the-wild robot policies. *Robotics: Science and Systems*, 2025.

[39] SAM 3D Team, Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, Aohan Lin, Jiawei Liu, Ziqi Ma, Anushka Sagar, Bowen Song, Xiaodong Wang, Jianing Yang, Bowen Zhang, Piotr Dollár, Georgia Gkioxari, Matt Feiszli, and Jitendra Malik. Sam 3d: 3dfy anything in images. 2025. URL https://arxiv.org/abs/2511.16624.

[40] Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 201–221. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/wang23a.html.

[41] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. FoundationPose: Unified 6d pose estimation and tracking of novel objects. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[42] Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. Foundationstereo: Zero-shot stereo matching. *arXiv*, 2025.

[43] Mengda Xu, Han Zhang, Yifan Hou, Zhenjia Xu, Linxi Fan, Manuela Veloso, and Shuran Song. Dexumi: Using human hand as the universal manipulation interface for dexterous manipulation. In *Conference on Robot Learning*, 2025.

[44] Hao Yang, Zhe Tao, Jian Yang, Wenpeng Ma, Haoyu Zhang, Min Xu, Ming Wu, Shuaishuai Sun, Hu Jin, Weihua Li, et al. A lightweight prosthetic hand with 19-dof dexterity and human-level functions. *Nature Communications*, 16(1):955, 2025.

[45] Jessica Yin, Haozhi Qi, Youngsun Wi, Sayantan Kundu, Mike Lambeta, William Yang, Changhao Wang, Tingfan Wu, Jitendra Malik, and Tess Hellebrekers. OSMO: Open-Source Tactile Glove for Human-to-Robot Skill Transfer. *arXiv preprint arXiv:2512.08920*, 2025.

[46] Zhao-Heng Yin, Changhao Wang, Luis Pineda, Fran-

cois Hogan, Krishna Bodduluri, Akash Sharma, Patrick Lancaster, Ishita Prasad, Mrinal Kalakrishnan, Jitendra Malik, et al. Dexteritygen: Foundation controller for unprecedented dexterity. *arXiv preprint arXiv:2502.04307*, 2025.

[47] Kelin Yu, Yunhai Han, Qixian Wang, Vaibhav Saxena, Danfei Xu, and Ye Zhao. Mimictouch: Leveraging multi-modal human tactile demonstrations for contact-rich manipulation. In *Conference on Robot Learning*, pages 4844–4865. PMLR, 2025.

[48] Chi Zhang, Penglin Cai, Haoqi Yuan, Chaoyi Xu, and Zongqing Lu. Unitachand: Unified spatio-tactile representation for human to robotic hand skill transfer. *arXiv preprint arXiv:2512.21233*, 2025.

[49] Han Zhang, Songbo Hu, Zhecheng Yuan, and Huazhe Xu. Doglove: Dexterous manipulation with a low-cost open-source haptic force feedback glove. *Robotics: Science and Systems*, 2025.

[50] Jialiang Zhao, Yuxiang Ma, Lirui Wang, and Edward Adelson. Transferable tactile transformers for representation learning across diverse sensors and tasks. In *Conference on Robot Learning*, pages 3766–3779. PMLR, 2025.

[51] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *Robotics: Science and Systems*, 2023.

APPENDIX A
DATASET

A. Wearable Devices

The human hand is a remarkably dexterous manipulator, with up to 23 degrees of freedom [44]. In contrast, robotic manipulators are undergoing rapid evolution: recent platforms exhibit increasing degrees of freedom alongside a growing diversity of mechanical designs and actuations [36, 14, 35]. The combination of fast hardware advancement and heterogeneous robot embodiments poses a challenge for data-driven policy learning, as demonstrations collected on any single platform risk becoming quickly outdated or narrowly tailored to a specific design. This motivates the need for a more generalizable source of dexterous data that is not tied to a particular robot. In this work, we therefore propose to treat the human hand as a universal embodiment for data collection, viewing human demonstrations as a shared data source that could, in principle, be leveraged across a wide range of robot hands as in [38, 45]. This approach differs from other hardware interfaces such as exoskeleton gloves (e.g., DOGLove [49], Dexop [9], DexUMI [43] in Fig. 11), which constrain natural hand motion and tend to produce data that is specific to a single robotic embodiment.

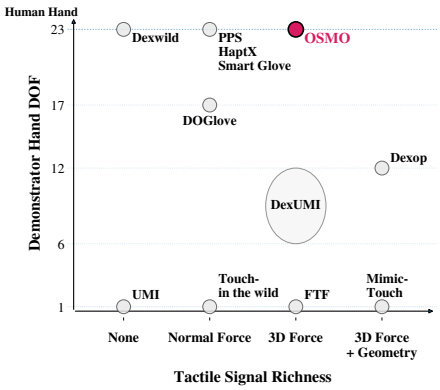


Fig. 11: Our human dataset leverages OSMO [45] tactile glove, providing the human demonstrator with full dexterity while capturing both shear and normal tactile signals.

B. Policy Dataset

For both the pivoting and insertion tasks, we reuse the 'seen-to-both' object demonstrations from the cross-sensor tactile alignment dataset.

C. Force Prediction Data-collection Setup

Figure 12 illustrates our force data collection setup. An ATI Gamma force-torque sensor is rigidly mounted to the tabletop to provide a stable reference frame for all interactions. During data collection, we align both the Xela tactile sensor and the glove fingertip along the +x direction of the sensor frame to ensure consistent contact orientation across trials.

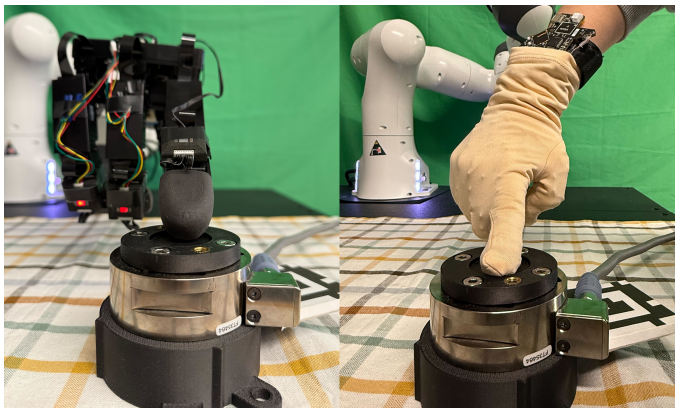


Fig. 12: Robot and human force label collection using an F/T sensor (ATI Gamma) fixed beneath the table.

D. Force Prediction Dataset

The robot dataset's train and test sets were randomly sampled from data whose ground-truth forces are approximately centered around the mean of (–0.0185, 0.1003, –2.1348) in the (x, y, z) directions. Similarly, the glove dataset's train and test sets were randomly sampled from forces centered around the mean of (0.1347, –0.1228, –1.5903). Overall, the mean x- and y-forces of the glove and robot datasets are comparable, whereas the robot dataset exhibits an approximately 0.5 N larger mean force in the z-direction.

E. H2R Policy Co-training Objects

Fig. 14 illustrates the evaluation objects and their corresponding names used in Tab. II. We also visualize the object categories using color-coded bounding boxes. As shown in the figure, we intentionally select objects with diverse geometries and physical properties to assess generalization across a broad range of manipulation settings.

APPENDIX B
POST PROCESSING

A. Data Post Processing

In this work, we extract fingertip locations and wrist poses using WiLoR [27], with hand masks obtained from SAM3 [5]. To improve hand pose accuracy, we refine the estimated hand meshes using ICP [31] with depth maps from FoundationStereo [42], and apply a Savitzky–Golay filter [33] for temporal consistency.

For object pose estimation, we propose a mesh-less object pose extraction pipeline adapted from Any6D [20]. We use FoundationPose [41] for object pose estimation on depth maps from FoundationStereo [42]. To run FoundationPose, we perform automatic mesh reconstruction once per task for both the robot and glove datasets, using the first frame of the first human demonstration $\mathcal{T}_1^h$. Object meshes are automatically generated from RGB images using an object name prompt via SAM3 [5] and SAM3D [39]. SAM3 [5] segments the object from the RGB image, and SAM3D reconstructs textured object meshes from the segmented object, even under occlusions.

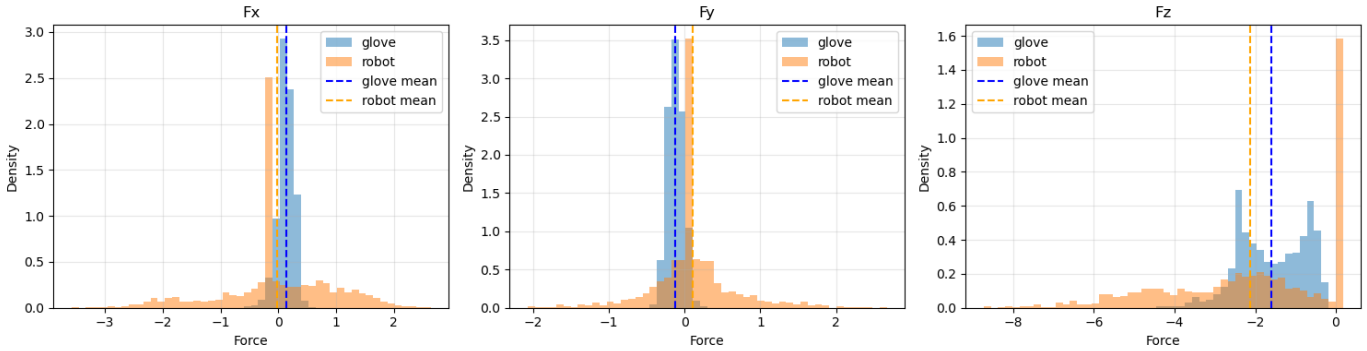


Fig. 13: Force dataset distribution



Fig. 14: Red boxes denote 'Seen-by-all' objects, blue boxes denote 'Human-only' objects, and green boxes denote 'Held-out' objects for each task in Tab. II.

With FoundationPose [41], we estimate the object pose directly on the initial frame of each trajectory, and then switch to tracking mode for subsequent frames to ensure temporal consistency. Although the objects are symmetric, SAM's texture generation helps disambiguate symmetry when used with FoundationPose (e.g., for the Cheez-It box). For objects with symmetric adapter textures, we use the initial object pose from the first human trajectory ($\mathcal{T}_1^h$) to align the object axes, assuming that objects share approximately the same initial state as specified in the problem statement.

Our entire pipeline is executed sequentially in a single pass and without the use of any privileged information, such as pre-existing 3D CAD models. We demonstrate this capability on pivoting with a box (Cheez-It) and insertion with an adapter (RVP+).

B. Signal Post Processing

Human For the human glove data, we first estimate and remove the baseline by measuring the non-contact state at the beginning and end of each trajectory and subtracting it from the raw signal. The current OSMO [45] glove exhibits sign flips depending on the underlying magnetic skin quadrant under the same applied force; to avoid ambiguity in contact location, we take the absolute value of the tactile signal. Glove data are collected at 30 fps.

Robot: Robot tactile data are collected at 100 fps; since the Xela sensor signal drifts over time, we record a baseline before every data collection session or robot rollout and subtract it accordingly.

Appendix C
Pseudo-pairs

A. Pose Normalization

For each task τ, we compute normalization statistics from the robot dataset and apply the same normalization to both human and robot data belonging to that task. Specifically, we subtract the mean and divide by the standard deviation, where the standard deviation is taken as the maximum across axes. This normalization is performed separately for position and orientation.

1) Fingertip position normalization: Let

$$\mathcal{P}_\tau^r = \{p_{t,k}^r \mid (F_t^r, P_t^r, w_t^r, o_t^r) \in A_\tau^r\}$$

denote the set of robot fingertip positions for task τ. We compute the mean from the robot data as

$$\mu_p(\tau) = \frac{1}{|\mathcal{P}_\tau^r|} \sum_{p \in \mathcal{P}_\tau^r} p, \tag{5}$$

and the maximum per-axis standard deviation as

$$\sigma_{p,\max}(\tau) = \max_{d\in\{x,y,z\}} \mathrm{std}(\{p_d : p \in \mathcal{P}_\tau^r\}). \tag{6}$$

We then normalize each fingertip position for *both* embodiments as

$$\tilde{p}_{t,k}^e = \frac{p_{t,k}^e - \mu_p(\tau)}{\sigma_{p,\max}(\tau)}, \qquad e \in \{h, r\}.$$

2) Object pose normalization: We write the object pose as $o_t^e = (s_t^e, q_t^e)$, where $s_t^e \in \mathbb{R}^3$ is position and $q_t^e \in \mathbb{R}^3$ is a rotation vector. Define the set of robot object positions for task τ as

$$\mathcal{S}_\tau^r = \{s_t^r \mid (F_t^r, P_t^r, w_t^r, o_t^r) \in A_\tau^r\}.$$

We compute the mean from the robot data as

$$\mu_s(\tau) = \frac{1}{|\mathcal{S}_\tau^r|} \sum_{s\in\mathcal{S}_\tau^r} s,$$

and the maximum per-axis standard deviation as

$$\sigma_{s,\max}(\tau) = \max_{d\in\{x,y,z\}} \mathrm{std}(\{s_d : s \in \mathcal{S}_\tau^r\}).$$

We normalize object positions for both embodiments as

$$\tilde{s}_t^e = \frac{s_t^e - \mu_s(\tau)}{\sigma_{s,\max}(\tau)}, \qquad e \in \{h, r\}.$$

For orientations, represented as rotation vectors, we apply the same procedure. Define the set of robot orientations for task τ as

$$\mathcal{Q}_\tau^r = \{q_t^r \mid (F_t^r, P_t^r, w_t^r, o_t^r) \in A_\tau^r\}.$$

We compute the mean from the robot data as

$$\mu_q(\tau) = \frac{1}{|\mathcal{Q}_\tau^r|} \sum_{q\in\mathcal{Q}_\tau^r} q,$$

and the maximum per-axis standard deviation as

$$\sigma_{q,\max}(\tau) = \max_{d\in\{x,y,z\}} \mathrm{std}(\{q_d : q \in \mathcal{Q}_\tau^r\}).$$

We normalize object orientations for both embodiments as

$$\tilde{q}_t^e = \frac{q_t^e - \mu_q(\tau)}{\sigma_{q,\max}(\tau)}, \qquad e \in \{h, r\}.$$

B. Balancing Term

In all experiments, we use a fixed balancing term of $\lambda = 1$ across all tasks and embodiments. To assess the sensitivity of our method to this choice, we report the alignment performance under neighboring values of λ. Table V shows the EMD reduction rate before and after alignment for different λ. We observe that the performance remains stable over a reasonably wide range of λ, indicating that our method is not sensitive to this hyperparameter.

λ	0.8	0.9	1.0	1.1	1.2
EMD Red. [%] ↑	80.5	82.5	83.2	80.6	79.2

TABLE IV: EMD reduction rate (EMD Red.) before and after alignment for different values of λ. Higher values indicate better alignment between human and robot tactile distributions. Performance remains robust across a range of λ.

C. Further Clarification of Pseudo-Pair Construction

In Sec. III-C, we define the pseudo-pair set P using a similarity threshold δ for notational compactness. In practice, δ primarily serves as a validity constraint in pseudo-pair construction, as we perform nearest-neighbor matching subject to this threshold. Specifically, for each human finger-object transition O_i^h, we select the N nearest robot transitions under the similarity metric $S(O_i^h, O_j^r)$. The threshold δ is then applied as a lower bound to exclude degenerate matches (e.g., those with large pose discrepancies). Intuitively, if δ is set too large, it can admit low-quality or semantically meaningless pairs into P. However, within a reasonable range, the resulting pseudo-pair set and consequently our alignment performance remains largely consistent.

To validate this, we perform a sensitivity analysis of the tactile alignment with respect to δ, as shown in Tab. V. The results indicate that δ can take a broad range of values without significantly affecting performance. All experiments in the main paper therefore use a fixed N and a single reasonable δ (with $N = 3$ and $\delta = 2.0$).

δ	1.5	2.0	2.5	3.0	3.5
EMD Red. [%] ↑	82.7	83.2	83.9	80.7	83.9

TABLE V: EMD reduction rate (EMD Red.) before and after alignment for different values of λ for different δ.

D. Binary Contact Threshold

To motivate our choice of non-contact thresholds, we visualize the distribution of raw tactile signal norms for both human and robot sensors in Fig. 15. In practice, non-contact signals tend to exhibit a high-density peak near zero, whereas when contact occurs, the signal magnitude typically changes significantly. This clear separation makes threshold-based detection of contact straightforward. We therefore select thresholds that are small enough to capture the dominant non-contact peak in the histogram while excluding meaningful contact signals. Specifically, we set $\delta_h = 1200$ for the human glove and $\delta_r = 30$ for the robot Xela sensor.

APPENDIX D
TRAINING AND ARCHITECTURE DETAILS

A. Alignment Module

In our implementation, we parameterize v_θ using a compact multilayer perceptron with three hidden layers of width 1024. Following [23], we train the model with 100 discretized time

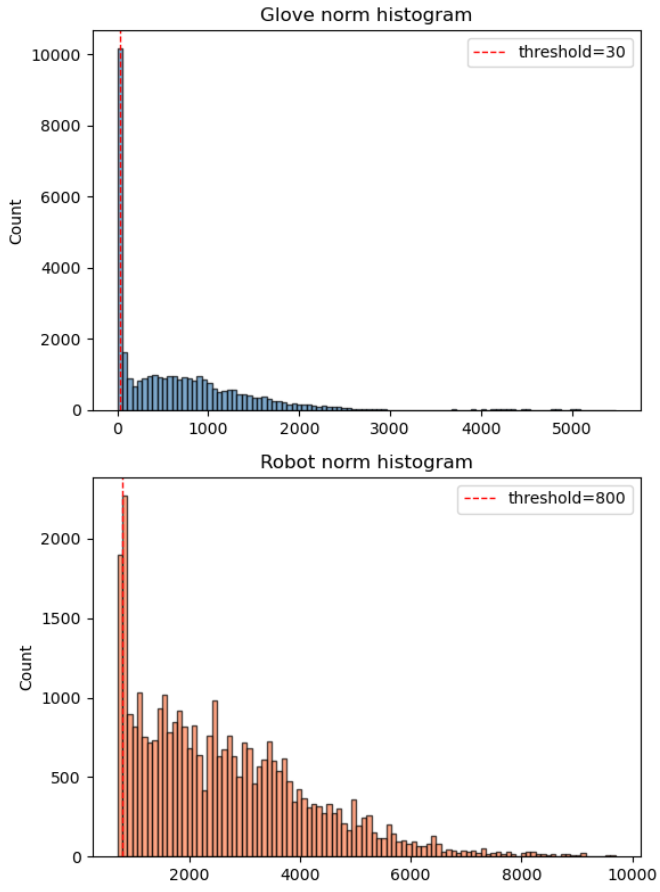


Fig. 15: Norm of raw sensor signal observations from the human glove [45] and the robot Xela sensor. The red dotted vertical line indicates the chosen non-contact threshold.

steps for 200,000 epochs using a learning rate of 5×10^{-5}. In our case, the entire training procedure takes about 10 minutes on a single NVIDIA RTX 4090.

B. H2R Policy – Pivoting

For pivoting, the policy takes as input only the index finger's tactile and proprioceptive signals. We treat the index fingertip as a representative of the many fingers involved in manipulation. The action space is 6-dimensional, corresponding to the index fingertip position and wrist rotation, and both the input and action spaces are defined with respect to the robot base frame.

C. H2R Policy – Insertion

For insertion, only three fingers (thumb, index, and middle) are actively engaged. Accordingly, the policy input consists of the tactile and proprioceptive signals from these three fingers. The 6-dimensional action output represents the index fingertip position and wrist rotation (expressed in the robot base frame), as this task leverages only wrist motion.

D. H2R Policy – Lid Closing

For lid closing, all four fingers (thumb, index, middle, and ring) are engaged. Thus, the policy input includes tactile and proprioceptive signals from all four fingertips. Similar to insertion, this task also leverages only wrist motion, and the 6-dimensional action output corresponds to the index fingertip position and wrist rotation, defined in the robot base frame.

E. Force Decoder

The force decoder is implemented as a multilayer perceptron with two hidden layers of 64 units each and ReLU activations. It is trained using an ℓ_2 regression loss with a learning rate of 10^{-4} for 100k epochs.

F. H2R Policy – Lightbulb Screwing

For lightbulb screwing, the policy uses all four fingertip positions as both input and output, expressed in the robot base frame. In this case, we do not include wrist rotation in the action space, as the wrist remains stationary throughout the interaction.

G. Learned Rectified Flow

Fig. 16 shows how the glove latent evolves to match the robot distribution. This figure shares the same color coding as Fig. 5.

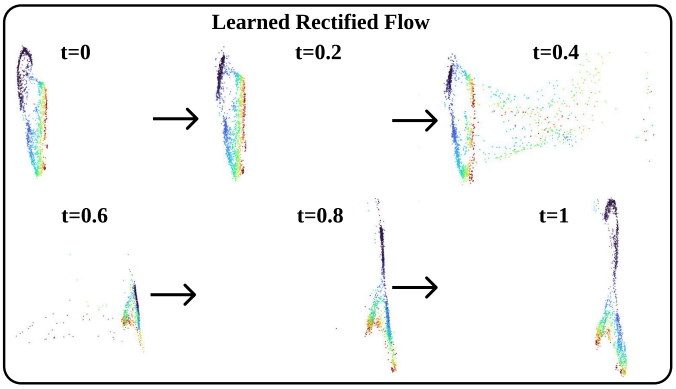


Fig. 16: How the tactile latent distribution of human's evolve with time via rectified flow.

APPENDIX E
INFERENCE

A. Policy Rollouts

During inference, the policy operates at 10 Hz with an action chunk size of 32. For pivoting, we execute 4 actions per rollout; for insertion, we execute 2 actions; and for lid closing, we execute 8 actions. During rollouts, the predicted action representations are converted into a sequence of joint commands via inverse kinematics, solved using PyRoki [18]. For all co-training tasks with wrist-only actions, we convert the predicted index-fingertip location and wrist orientation into a wrist pose in $\mathrm{SE}(3)$ by assuming a rigid transformation between the wrist and the index fingertip.

For light-bulb screwing, the policy runs at a higher frequency of 30 Hz, and we execute 12 actions from the 32-step action chunk to enable fine-grained manipulation at a higher control rate.