CONACT: CONTACT AS ACTION FOR MULTI-FINGERED ROBOT HAND POLICIES FROM HUMAN VIDEOS

Anonymous authors
Paper under double-blind review

ABSTRACT

Policies for multi-fingered robot hands are typically trained on teleoperated demonstrations, which are slow and costly to collect, especially for bimanual tasks. Human videos offer a scalable alternative, yet they show where the hands move but not how firmly they press. We argue that this missing contact is a key reason why policies pretrained on human videos still rely on many teleoperated demonstrations. To address this limitation, we introduce **ConAct** (Contact as Action), a bimanual policy for multi-fingered hands that learns from human videos with few teleoperated demonstrations. Specifically, our key idea is to extend the point-based action representation shared by human and robot hands with contact, *i.e.*, a contact intensity at each 3D keypoint, which we label automatically on about 1M egocentric video episodes without any tactile sensor or manual annotation. This design not only lets the policy learn from human videos how firmly to press, but also lets us turn the predicted contact into force along the normals of the robot hand at deployment, without any force sensing. On a real bimanual robot with two 20-DoF hands, ConAct outperforms three Vision-Language-Action models pretrained on human videos on all three tasks at every amount of teleoperated demonstrations. For example, with only 20 teleoperated demonstrations in total, it more than doubles the average success rate of the strongest baseline ($17.2\% \rightarrow 41.7\%$) and already matches a baseline trained with ten times more. Our ablations further show that both the contact intensity and the force it drives contribute substantially to this gain. Moreover, on a bimanual task, ConAct more than quadruples the success rate of the best baseline with only five teleoperated demonstrations ($12.5\% \rightarrow 53.1\%$). Videos are available at https://anonymous39306.github.io/conact.

1 INTRODUCTION

Unlike text and images, robot trajectories cannot be scraped from the web. Every demonstration must be physically executed, typically through teleoperation, and even the largest open robot datasets (Open X-Embodiment Collaboration, 2024) remain orders of magnitude smaller than the corpora behind language and vision models. This scarcity is most acute for multi-fingered hands, whose action dimensionality far exceeds that of parallel-jaw grippers, so each demonstration is slower and costlier to collect, and bimanual tasks double the burden.

To address this scarcity, a growing line of work mines the vast pool of human videos available online (Grauman et al., 2022) as a scalable substitute. For example, prior works learn visual or reward representations from human videos (Nair et al., 2022), extract coarse action priors from hand motion (Bahl et al., 2023), or pretrain Vision-Language-Action (VLA) models on egocentric videos, *e.g.*, with hand pose as the action (Li et al., 2026) or with an action space shared between human and robot (Llontop et al., 2026). However, these VLAs still adapt to a new multi-fingered platform with 100 or more teleoperated demonstrations per task.

We argue that a key reason is what human videos leave out. Specifically, they show where the hands move but not how firmly they press. In teleoperated demonstrations, by contrast, the commanded finger joints close past the object surface, and the resulting tracking error produces the grip force. Hence, these demonstrations carry contact implicitly. For example, DexWild (Tao et al., 2025) reports that policies trained on its human data alone reach only 3.6% in-domain success, and notes that relying only on visual and kinematic data limits the performance in contact-rich tasks.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

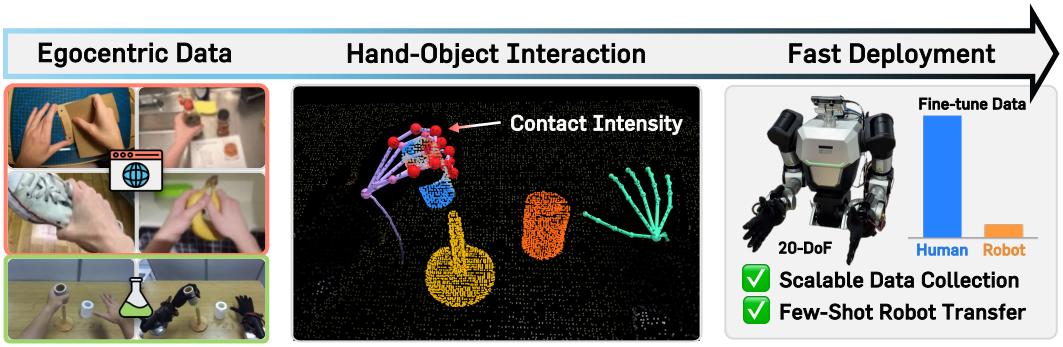


Figure 1: **ConAct.** (Left) We learn from egocentric human videos, *i.e.*, a large pretraining corpus and our own demonstrations. (Middle) From each video, we extract the 3D points of the hands and the objects together with a contact intensity at each hand keypoint, which is labeled automatically without any tactile sensor. The policy predicts these points and contact intensities as an action shared by human and robot hands. (Right) Human videos provide most of the fine-tuning data, and ConAct transfers to a bimanual robot with multi-fingered hands from few teleoperated demonstrations.

Meanwhile, point-based representations make the kinematic part of this transfer cheap. They describe the scene and the hand by 3D points, *e.g.*, on the object surfaces and at the fingertips, rather than by images or robot-specific joint angles (Haldar & Pinto, 2025). Since these points live in the same metric space for human and robot hands, a human trajectory directly serves as a target for the robot, and recent works extend this idea to multi-fingered hands (Guzey et al., 2026; Wu et al., 2026). However, the points describe only where the hand goes, so these works close the fingers by fixed rules and name the missing force as a limitation. Other works obtain contact from physics simulation (Mu et al., 2026) or from tactile gloves and robot tactile sensors (Zhang et al., 2026a). Hence, learning from ordinary human videos how firmly the hand presses remains open.

In this paper, we introduce **ConAct** (Contact as Action), a bimanual policy for multi-fingered hands that learns from human videos with few teleoperated demonstrations. Our key idea is to treat contact as an action shared by human and robot hands, and our contributions are threefold. First, we extend point-based representations with a contact intensity at each of the 21 MANO keypoints (Romero et al., 2017) of each hand, so that they specify how firmly each keypoint presses in addition to where it goes (Figure 1). Second, we supervise this contact at scale without any tactile sensor or manual annotation. Specifically, we label about 1M egocentric episodes and our human demonstrations automatically with a recent video-based hand tracker (Anonymous, 2026), so that the policy learns contact from pretraining onward. Third, we turn the predicted contact into force along the normals of the robot hand without hand-specific grip heuristics, and deploy the policy on a real bimanual robot (Rainbow Robotics, 2024) with two 20-DoF hands (Wuji Technology, 2026).

We evaluate ConAct on this robot across one unimanual and two bimanual tasks, and compare it against three VLAs pretrained on human videos, *i.e.*, VITRA (Li et al., 2026), EgoVLA (Yang et al., 2026a), and GR00T N1.7 (Llontop et al., 2026). Every method is fine-tuned on the same human demonstrations plus 10%, 50%, or 100% of the teleoperated ones. Notably, the 200 teleoperated demonstrations took expert operators 3 person-hours, whereas as many human demonstrations took only 0.5, *i.e.*, about one sixth. ConAct outperforms every baseline on all three tasks at every amount of teleoperation data. For example, with only 20 teleoperated demonstrations in total, it more than doubles the average success rate of the strongest baseline ($17.2\% \rightarrow 41.7\%$) and already matches a baseline trained with ten times more. On a bimanual task, it more than quadruples the success rate of the best baseline with only five teleoperated demonstrations ($12.5\% \rightarrow 53.1\%$). Our ablations show that both the contact intensity and the force it drives contribute substantially to these gains.

Overall, our work highlights that human videos can teach robot hands not only where to move but also how firmly to press, and we believe it could inspire further research on human-to-robot transfer.

2 Related Work

2.1 Robot Policy Learning from Human Videos

Learning robot policies from human videos has been actively studied to reduce the need for robot data. For example, prior works learn visual representations from egocentric videos (Nair et al., 2022; Ma et al., 2023; Karamcheti et al., 2023; Majumdar et al., 2023), or extract affordances, point trajectories, and hand motions from human videos to guide imitation learning (Bahl et al., 2023; Bharadhwaj et al., 2024; 2025; Wen et al., 2024; Mandikal & Grauman, 2022; Shaw et al., 2022). Building on a long line of keypoint representations for manipulation (Manuelli et al., 2019; Simeonov et al., 2022; Huang et al., 2024; Liu et al., 2024; Yuan et al., 2024b; Levy et al., 2025), point-based policies (Haldar & Pinto, 2025; Haldar et al., 2026) express both the observation and the action as 3D points shared by the human hand and the robot gripper, and recent works extend them to multi-fingered hands by predicting fingertip trajectories (Guzey et al., 2026) or static grasps (Wu et al., 2026). Other works close the gap by inpainting the robot into human videos (Lepert et al., 2025) or by converting human demonstrations into robot data in simulation (Mu et al., 2026; Yang et al., 2026b; Chen et al., 2026; Ren et al., 2026; Zhu et al., 2026). We treat point representations, including trajectory sketches (Gu et al., 2024) and 3D flow (Yuan et al., 2024a), as prior work rather than as our contribution, and ConAct extends the point-based formulation to both hands with the full MANO layout and a contact intensity per keypoint. Most relevant to our setting are VLAs pretrained on large corpora of egocentric videos, which we compare against, rather than VLAs pretrained on robot data alone, *e.g.*, $\pi_{0.5}$ (Black et al., 2025), since ConAct likewise learns from human videos with few teleoperated demonstrations.

Among the former, **VITRA** (Li et al., 2026) pretrains a VLA on a large corpus of egocentric human hand videos, treating the robot hand action space as a subspace of the human one, but adapts to a robot with many teleoperated demonstrations, *e.g.*, 2610 in their setting. We pretrain on the same corpus, but condition a diffusion transformer on point tokens instead of a vision-language backbone. **EgoVLA** (Yang et al., 2026a) and Being-H0 (Luo et al., 2026) likewise pretrain VLAs on egocentric human videos with hand pose as an intermediate action, and adapt to a robot with teleoperated demonstrations. **GR00T N1.7** (Llontop et al., 2026) pretrains on robot data together with the 20K hours of egocentric human videos of EgoScale (Zheng et al., 2026), using a relative end-effector action space shared between human and robot, and is the closest large-scale counterpart to our setting. We compare against the three models in bold, fine-tuned on the same human and teleoperated demonstrations (Section 4.1).

2.2 Contact and Force from Human Videos

In computer vision, prior works detect whether a hand is in contact in internet images (Shan et al., 2020; Narasimhaswamy et al., 2020), recover grasp contact on object surfaces (Brahmbhatt et al., 2020; Grady et al., 2021), estimate hand pressure from RGB images (Grady et al., 2022; 2024; Zhao et al., 2025), and infer the forces of human–object interactions from monocular videos (Li et al., 2019; Ehsani et al., 2020). Most recently, tactile signals over the whole hand are predicted from egocentric videos with models trained on tactile-glove recordings (Zhang et al., 2026b; Zhou et al., 2026). However, these works treat contact as a perception output rather than as something a robot executes. We build on this line of work through a recent hand tracker (Anonymous, 2026), which performs hand-mesh recovery (Pavlakos et al., 2024; Zhang et al., 2025) with a video foundation model and also estimates contact over the hand surface, and use it to label contact in every human video we train on (Section 3.1).

In robot learning, force from humans is instead recorded with tactile gloves or exoskeletons for robots equipped with tactile sensors (Adeniji et al., 2025; Zhang et al., 2026a; Xu et al., 2025), which limits the data to what such devices collect. Other policies use contact in different forms, *e.g.*, as the point a gripper should touch (Cui et al., 2026), as a target for pretraining visual features (Gavryushin et al., 2025), or as a force goal predicted from images (Collins et al., 2024). Meanwhile, point-based hand policies without robot data close the fingers by fixed rules (Guzey et al., 2026; Wu et al., 2026), and simulation-based methods obtain contact from grasp optimization or physics (Mu et al., 2026; Yang et al., 2026b). In contrast, ConAct learns from ordinary videos how firmly each keypoint presses and turns the predicted contact into force on robot hands without tactile sensing.

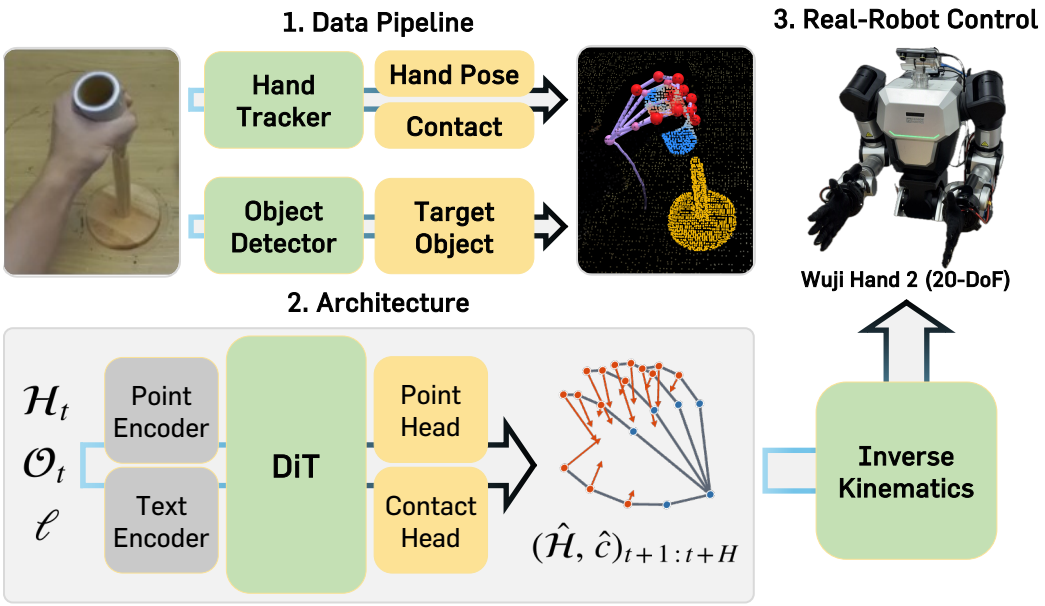


Figure 2: **Overview.** (1) Point and contact extraction: from an egocentric frame and a task description, we extract *object points* with text-prompted segmentation and depth. Then, a hand tracker reconstructs the mesh of each hand, from which we take the *21 MANO keypoints* and a *contact intensity* per keypoint, without any tactile sensor. (2) Architecture and training: a Sentence Transformer encodes the language instruction and the object names, PointNet the object points, and an MLP the keypoints of both hands into tokens. Then, a diffusion transformer attends to them together with H action tokens and generates the next H keypoint positions $\hat{\mathcal{H}}_{t+1:t+H}$ and contact intensities $\hat{c}_{t+1:t+H}$ of both hands, which a keypoint head and a contact head produce from the same tokens, in pretraining and fine-tuning alike. (3) Deployment: the predicted contact displaces each keypoint target along the direction in which the finger closes, and inverse kinematics maps the displaced targets to the joint targets of the arm and the hand, which turns contact into force on the robot.

3 ConAct: Contact as Action

ConAct is a bimanual policy for multi-fingered hands that treats contact as an action shared by human and robot hands. Specifically, the policy observes the 3D points of the task-relevant objects and the 21 MANO keypoints of each hand, and predicts the future keypoints together with a contact intensity at each of them (Figure 2). In this section, we first describe how we extract the keypoints and their contact from videos without any tactile sensor (Section 3.1). Then, we describe how we train a diffusion transformer to predict them (Section 3.2) and how we turn the predicted contact into force on a robot hand (Section 3.3).

3.1 Data

Human videos. We pretrain on the VITRA corpus (Li et al., 2026), which provides about 1M egocentric episodes with language captions and hand tracks. For fine-tuning, we record task-specific demonstrations with an ego-view ZED Mini stereo camera and the operator's bare hands, which is substantially faster than teleoperation, *i.e.*, 0.5 person-hours for our 200 demonstrations versus 3 person-hours of expert teleoperation for the same number (Appendix B).

Object points. We segment each task-relevant object by its name with a text-prompted video segmenter and lift 512 points of its mask to 3D with stereo depth (monocular depth for the pretraining corpus), which gives the object points at every frame (Appendix B).

Hand keypoints and contact. We represent each hand, human or robot, by the 21 MANO keypoints (Romero et al., 2017), each with a 3D position and a contact intensity. We need contact because

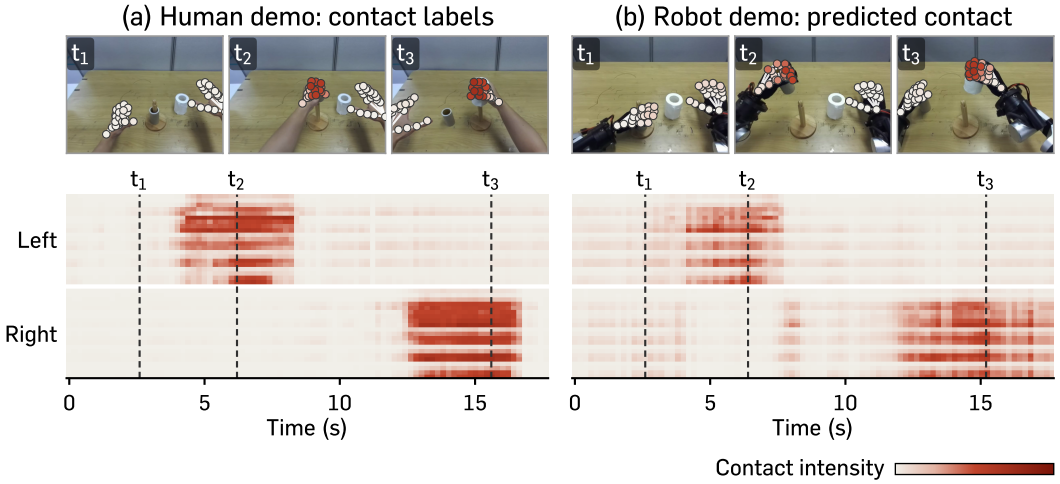


Figure 3: **Contact from human videos to robot hands.** (a) Contact intensities labeled by the hand tracker on one of our human demonstrations of Replace Tissue Roll, aggregated to the 21 keypoints of each hand. (b) Contact intensities predicted by ConAct on the robot hands for the same task, which drive the force injection (Section 3.3).

a point-only policy predicts where the hand goes but not how firmly it presses. Specifically, once contact is established, hand and object points stop moving even as the hand continues to apply force, so point trajectories alone cannot distinguish a light touch from a firm grasp. We obtain both from video with a recent hand tracker (Anonymous, 2026), which jointly predicts the hand mesh and a contact probability at each of its vertices. We take the keypoints from the mesh and aggregate the vertex probabilities around each keypoint. The aggregate grows with the contact area around the keypoint, so we interpret it as how firmly the keypoint presses, *i.e.*, its contact intensity, which gives a contact vector $c_t \in [0,1]^{21}$ per hand (Figure 3a and Appendix A). For our demonstrations, we also align the scale of the hands to the stereo depth of the wrist at every frame (Appendix B). For the pretraining corpus, we keep the keypoints of VITRA's hand tracks and add the contact of the tracker. On the robot, the keypoints are sites on the hand model, computed by forward kinematics.

3.2 Policy Training

Architecture. We instantiate the policy as a diffusion transformer (DiT) (Peebles & Xie, 2023). It observes the language instruction that specifies the task, the names of up to four task-relevant objects, and, from the current frame, the points of these objects and the keypoints of both hands, each encoded into tokens by a lightweight encoder (Appendix C). We note that contact is not part of the observation, so the robot needs no contact sensing. The DiT takes these observation tokens followed by $H = 16$ action tokens, one per future step, and from each action token, a keypoint head and a contact head predict the keypoints of both hands and their contact intensities,

$$a_{t+h} \;=\; \big[\hat{\mathcal{H}}_{t+h},\; \hat{c}_{t+h}\big], \qquad \hat{\mathcal{H}}_{t+h} \in \mathbb{R}^{42\times 3},\;\; \hat{c}_{t+h} \in [0,1]^{42}. \tag{1}$$

Objective. The keypoint head is trained with rectified flow (Liu et al., 2023; Lipman et al., 2023), *i.e.*, it predicts the velocity $\hat{v}$ that transports Gaussian noise ϵ toward the target chunk d along $d_\tau = (1-\tau)\,\epsilon + \tau\, d$, where d holds the displacements of the keypoints from their current positions, and the generated displacements are added to the current keypoints to give $\hat{\mathcal{H}}_{t+1:t+H}$. The contact head is trained with a binary cross-entropy against the continuous intensity labels, without a positive-class weight, so that it regresses the intensity:

$$\begin{aligned} \mathcal{L} \;&=\; \mathcal{L}_{\text{act}} + \lambda\, \mathcal{L}_{\text{ct}}, \qquad \mathcal{L}_{\text{act}} \;=\; \mathbb{E}_{\tau,\epsilon}\big\|\hat{v}(d_\tau,\tau) - (d-\epsilon)\big\|_2^2, \\ \mathcal{L}_{\text{ct}} \;&=\; \frac{1}{H\,K}\sum_{h=1}^{H}\sum_{k=1}^{K} \mathrm{BCE}\big(\hat{c}_{t+h,k},\, c_{t+h,k}\big), \end{aligned} \tag{2}$$

with $K = 42$ and $\lambda = 1$. We detach the input of the contact head, so that the contact loss does not change the predicted keypoints and contact affects the robot only at deployment (Section 3.3).

Pretraining and fine-tuning. We pretrain the policy from random initialization on the VITRA corpus with the contact labels of the hand tracker, and fine-tune it with the same objective as a single policy for all three tasks, on the human demonstrations of all tasks and 10%, 50%, or 100% of the teleoperated demonstrations of each task. The teleoperated demonstrations are converted to the same representation, *i.e.*, their keypoints come from forward kinematics of the measured joint states and their object points from the same pipeline. They carry no contact labels, so the contact loss is masked on them. Hands absent from a clip, including the unused hand of our unimanual task, are masked out of both losses (Appendix C). Training details are in Appendix D.

3.3 Policy Inference

Perception. At deployment, we segment the task-relevant objects by their names and lift their points to 3D with stereo depth, as for our demonstrations, and compute the robot's hand keypoints by forward kinematics at the 21 mapped sites of each hand. All points are expressed in the camera frame, as in training (Appendix C).

Contact-force injection. Point targets cannot express how firmly to press. Therefore, we realize contact by displacing the targets of inverse kinematics (IK). Specifically, for each robot hand, we map the 21 keypoints to sites on its kinematic model and estimate a unit normal $\mathbf{n}_k$ at each site from the flexion direction of the motor that drives it, *i.e.*, the direction in which the site moves as the finger closes toward an object (Appendix E). Then, we displace each predicted keypoint $\hat{\mathbf{x}}_k$ along its normal in proportion to its predicted contact intensity $\hat{c}_k$ (Figure 3b),

$$\tilde{\mathbf{x}}_k \;=\; \hat{\mathbf{x}}_k + \delta\, \hat{c}_k\, \mathbf{n}_k, \tag{3}$$

with a single scalar $\delta = 5\,\mathrm{cm}$ shared by all keypoints, tasks, and objects. In practice, the predicted intensities rarely exceed 0.5, so the displacement stays below 2.5 cm. Intuitively, a keypoint predicted to be in contact is commanded slightly beyond the object surface, and the position-controlled joints that track it press the hand into the object with a force that grows with the predicted intensity. Keypoints with no predicted contact keep their predicted positions. We note that the rule involves only the mapped sites and their normals, so it requires no hand-specific grip heuristics.

Execution. The policy generates a chunk of $H = 16$ steps, which the robot executes in full at 20 Hz before the policy re-plans from a new observation. At each step, IK converts the displaced targets $\tilde{\mathbf{x}}_{1:21}$ of each hand into joint targets for the arm and the hand, which we linearly interpolate to 100 Hz for the arms and 500 Hz for the hands. The IK weights the error of each keypoint more heavily as its predicted contact grows, so that the keypoints in contact are tracked most closely (*contact-weighted IK*, Appendix E).

4 Experiments

In this section, we demonstrate the effectiveness of ConAct on real-robot manipulation tasks. Specifically, we describe the experimental setup in Section 4.1, present the main results in Section 4.2, and ablate the contact design in Section 4.3. We aim to answer the following questions:

- **Q1.** How efficiently does ConAct use teleoperation data compared with VLA baselines?
- **Q2.** How much does contact as action contribute to this efficiency?
- **Q3.** Can ConAct generalize to scenes with a distractor?

4.1 Experimental Setup

Hardware. We deploy ConAct on an RB-Y1 with two Wuji hands (Wuji Hand 2), each with 20 actuated DoF (Appendix E). A ZED Mini stereo camera on the robot's head, the same model as for the human demonstrations, is mounted to approximate their egocentric viewpoint and provides rectified 1280×720 frames. Fast-FoundationStereo (Wen et al., 2026) computes depth from the stereo pair for the fine-tuning data and at deployment (Appendix B).

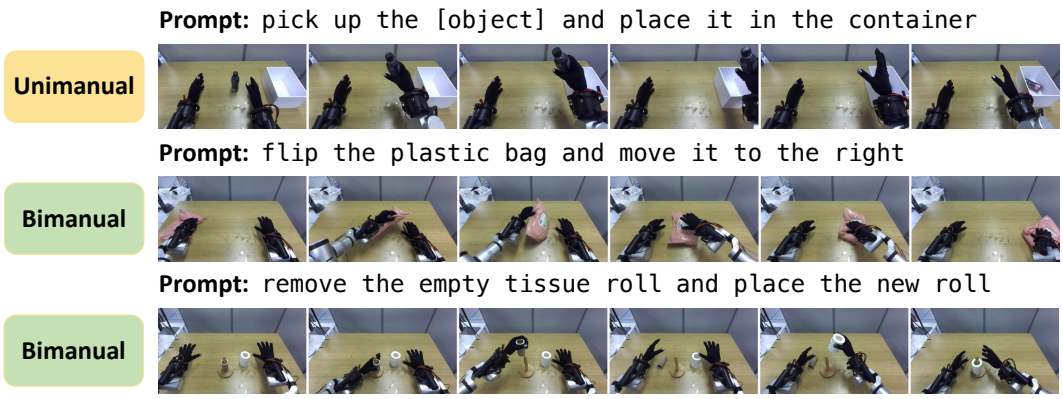


Figure 4: **Real-world rollouts.** Rollouts of a single ConAct policy on the three tasks with the RB-Y1 and two Wuji hands, each shown with the language prompt that specifies the task.

Tasks. We evaluate on three real-world tasks, one unimanual and two bimanual (Figure 4):

- **Pick & Place** (unimanual). The right hand picks up one of four objects, *i.e.*, a bottle, a box, a doll, and a ball, at one of 16 positions on a 4×4 grid and places it into a container.
- **Flip Plastic Bag** (bimanual). The left hand flips a plastic bag over, and then the right hand pushes it to a target region.
- **Replace Tissue Roll** (bimanual). The left hand pulls the empty core of a used-up tissue roll off a vertical holder, and then the right hand puts a new tissue roll on.

We collect 200 human demonstrations and 200 teleoperated demonstrations, split across tasks, *i.e.*, 100 for Pick & Place (25 per object) and 50 for each bimanual task. We train a single policy for all three tasks, which identifies the task from its language instruction (Figure 4). The bimanual tasks differ in coordination, *i.e.*, both hands act on the same object in turn in Flip Plastic Bag, whereas each hand handles a different object in Replace Tissue Roll. We provide the task setup and success criteria in Appendices F and G.

Data regimes and evaluation. We fix the human fine-tuning data, *i.e.*, every method is fine-tuned on the same in-house human demonstrations after its own pretraining, and vary only the amount of teleoperation data. Specifically, we fine-tune every method with 10%, 50%, or 100% of the teleoperated demonstrations of each task, with the same subsets for every method. For example, 10% amounts to 20 demonstrations in total, *i.e.*, 10 for Pick & Place and 5 for each bimanual task. We report the single-attempt success rate (%) over 16 trials per task and object, *i.e.*, 64 for Pick & Place with its four objects and 16 for each bimanual task, with the same initial configurations for every method. In bimanual tasks, partial success counts as 0.5 (Appendix G).

Baselines. We compare ConAct against three Vision-Language-Action (VLA) models that are likewise pretrained on human videos. Specifically, **VITRA** (Li et al., 2026) and **EgoVLA** (Yang et al., 2026a) are pretrained on egocentric human videos with hand pose as an intermediate action. **GR00T N1.7** (Llontop et al., 2026) is pretrained on large robot datasets and the 20K hours of egocentric human videos of EgoScale (Zheng et al., 2026) with a relative end-effector action space shared between human and robot. Like ConAct, we fine-tune every baseline from its public checkpoint as a single model for all three tasks, on the same human demonstrations converted to its action space, *i.e.*, with our IK solver or as MANO parameters, and on the same teleoperated demonstrations (Appendix I).

4.2 Main Results

Efficiency (Q1). As shown in Table 1, ConAct outperforms every baseline on all three tasks at every amount of teleoperation data, and its margin grows as the teleoperation data shrinks. Averaged over the tasks, it improves over VITRA, the strongest baseline, by a relative 13% with 100%

Table 1: **Success rate versus teleoperation data.** Success rate (%) of each method fine-tuned on the same human demonstrations plus 10%, 50%, or 100% of the teleoperated demonstrations of each task, *i.e.*, up to 100 for Pick & Place and 50 for each bimanual task, on the RB-Y1 with two Wuji hands (Appendix H). We denote **best** and second best values.

	Unimanual			Bimanual								
	Pick & Place			Replace Tissue Roll			Flip Plastic Bag			Average		
Method	10%	50%	100%	10%	50%	100%	10%	50%	100%	10%	50%	100%
VITRA	39.1	68.8	82.8	6.3	56.3	59.4	6.3	21.9	50.0	17.2	49.0	64.1
EgoVLA	14.1	42.2	64.1	12.5	15.6	50.0	12.5	15.6	43.8	13.0	24.5	52.6
GR00T N1.7	15.6	39.1	56.3	12.5	25.0	34.4	9.4	15.6	31.3	12.5	26.6	40.6
Ours	**56.3**	**87.5**	**89.1**	**53.1**	**62.5**	**75.0**	**15.6**	**28.1**	**53.1**	**41.7**	**59.4**	**72.4**

(64.1% → 72.4%), 21% with 50% (49.0% → 59.4%), and 142% with 10% (17.2% → 41.7%). In other words, ConAct matches the baselines with far fewer teleoperated demonstrations. For example, with 10%, it already exceeds GR00T N1.7 trained with all of them on average (41.7% vs. 40.6%), and with 50%, it exceeds every baseline trained with all of them on Pick & Place (87.5% vs. at most 82.8%) and on Replace Tissue Roll (62.5% vs. at most 59.4%). On Pick & Place, ConAct nearly saturates with 50% (87.5% vs. 89.1% with 100%), whereas VITRA still gains 14.0 points from the other half. Moreover, when the teleoperation data shrinks from 100% to 10%, ConAct retains 58% of its average success, whereas every baseline retains at most 31%.

Scarce teleoperation data (Q1). With 10% of the teleoperation data, ConAct attains 56.3% on Pick & Place, compared with 39.1% for VITRA, and matches GR00T N1.7 trained with all of the teleoperated demonstrations. The gap is even larger on Replace Tissue Roll, where ConAct attains 53.1%, whereas every baseline stays at or below 12.5%. On Flip Plastic Bag, the most challenging task, ConAct still leads with little data (15.6% vs. at most 12.5%). The baselines learn less efficiently from the human demonstrations, since none of them represents the hand by points that human and robot hands share. Instead, they predict joint angles or MANO parameters, which must be mapped from one hand to the other. EgoVLA suffers the most from this mapping, *i.e.*, its predicted MANO hands often retarget to implausible robot hand poses, so many of its failures come from the retargeting itself rather than from the predicted hand motion. ConAct shows no distinctive failure mode, and its failures concentrate on the tasks that are also hard for the baselines. For example, the plastic bag of Flip Plastic Bag is deformable, which makes its points harder to estimate, and its soft surface pushes back little, so the contact intensity on it is less well defined.

Contact versus teleoperation data (Q2). We argued that the missing contact is a key reason why policies pretrained on human videos still rely on many teleoperated demonstrations (Section 1). To test this, we deploy ConAct with $\delta = 0$ in Equation (3) (*w/o force injection*), *i.e.*, without the contact-driven displacement but with the contact-weighted IK. Since the contact head is detached, the contact loss does not change the predicted keypoints, so this isolates the contribution of the contact-driven force in Equation (3). With 10% of the teleoperation data, the contact-driven force raises the success rate on Pick & Place from 25.0% ($\delta = 0$) to 56.3% (Table 3), well above the 39.1% of VITRA, the strongest baseline.

Generalization (Q3). We evaluate ConAct and the baselines on Pick & Place with 10% of the teleoperation data in scenes *with a distractor*, which the demonstrations never show. Specifically, in each of 16 trials, we place the target object and a distractor object at random positions on the table, whereas the table holds only the target in training. As shown in Table 2, ConAct keeps its success rate of 56.3% from the training scene, whereas every baseline collapses. For example, VITRA drops from 39.1% to 6.3%, losing 84% of its success, and EgoVLA and GR00T N1.7 fail in every trial. We attribute this robustness to what ConAct observes. The baselines encode the raw image, so the distractor shifts their visual features away from those seen in training and can draw the hand to the wrong object. In contrast, the text-prompted segmenter keeps the distractor out of the observation of

Table 2: **Generalization.** Success rate (%) on Pick & Place with 10% of the teleoperation data in the training scene (from Table 1) and with an unseen distractor object. We denote **best** values.

Method	Training	w/ distractor
VITRA	39.1	6.3
EgoVLA	14.1	0.0
GR00T N1.7	15.6	0.0
Ours	**56.3**	**56.3**

Table 3: **Ablation study.** Success rate (%) on Pick & Place with 10% of the teleoperation data. *w/o contact labels* pretrains without contact loss, *binary labels* thresholds contact intensities, and *w/o force injection* sets $\delta = 0$. We denote **best** values.

Component	Variant	Success rate
Contact Pretrain	w/o contact label	43.8
Contact Intensity	w/o binary label	37.5
Contact Offset	w/o force inject	25.0
Ours (Full)		**56.3**

ConAct, which contains only the points of the named objects and the hand keypoints. As a result, the scene with a distractor looks to the policy much like the training scene.

4.3 Ablation Study

We ablate the contact design of ConAct on Pick & Place with 10% of the teleoperation data (Table 3). Every component contributes, *i.e.*, large-scale contact labels add 12.5 points, graded intensities 18.8 points, and force injection 31.3 points.

Contact labels. First, we pretrain without the contact loss (*w/o contact labels*), so the contact head learns contact only from our 200 human demonstrations instead of about 1M episodes. The predicted keypoints are unaffected, since the contact head is detached, yet the success rate drops to 43.8%, as the predicted contact is often late or missing. This shows the benefit of labeling contact automatically at the scale of the pretraining corpus. Next, replacing the contact intensities with binary labels in both pretraining and fine-tuning (*binary labels*) lowers the success rate to 37.5%. Binary contact tells the robot when to press but not how firmly, so the displacement jumps between zero and the full δ. As a result, the robot may close or open its hands prematurely, leading to suboptimal grasp poses or even squeezing small objects out of its grasp.

Force injection. Finally, force injection contributes the most, as the $\delta = 0$ deployment of Section 4.2 shows ($25.0\% \rightarrow 56.3\%$). With the displacement, the fingers press into the object and hold it firmly, whereas without it they stop at the surface, so the object often slips when the arm lifts it, especially the heavy bottle. Hence, beyond reaching the right points, a multi-fingered hand must press on the object, which contact enables.

5 Conclusion

In this paper, we propose ConAct, a bimanual policy for multi-fingered robot hands that learns from human videos with few teleoperated demonstrations. Our key idea is to treat contact as an action shared by human and robot hands, which we label automatically from ordinary videos and turn into force on the robot hand. On a real bimanual robot, ConAct outperforms Vision-Language-Action models pretrained on human videos on all three tasks at every amount of teleoperated demonstrations, and both the contact intensity and the force it drives contribute substantially to this gain. Overall, our work highlights that human videos can teach robot hands not only where to move but also how firmly to press, and we hope it inspires further research on learning contact-rich skills from human videos.

Limitations. First, the policy inherits the errors of its automatic labels. Second, since contact is estimated from videos rather than measured, the applied force is calibrated only indirectly, *i.e.*, through task success. Finally, ConAct relies on sensed depth and a fixed egocentric viewpoint, and our evaluation covers three tasks on one robot. Beyond these, soft objects such as the plastic bag push back only weakly, so contact on them is less well defined. Far from a dead end, we see this as a wide-open frontier, *i.e.*, learning from human videos how firmly to handle deformable objects.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods, cleaning and reformatting datasets, and assisting with translation. We have not used generative AI tools for proposing or refining hypotheses, developing theoretical models or conceptual frameworks, designing or providing feedback on research methodology or experiments, or interpreting results. Generating synthetic data sets, formulating mathematical claims, providing ingredients for or writing proofs, and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for drafting parts of the research paper. We have reviewed all AI-assisted work. The implemented methods are the model and training code, and the reformatted datasets are the human and robot demonstrations that our pipelines convert into point trajectories; two authors verified and tested this code for correctness. For the manuscript, AI tools assisted translation and drafting, and the authors reviewed, revised, and finalized every passage of the paper and its appendix. Three authors carried out and verified every analysis, including recomputing each reported success rate from the evaluation logs. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

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. 3

Anonymous. Video foundation models as physically grounded hand trackers for robot learning. Under review, 2026. URL https://anonymous4898.github.io/rlhnd/. 2, 3, 5, 15

Shikhar Bahl, Russell Mendonca, Lili Chen, Unnat Jain, and Deepak Pathak. Affordances from human videos as a versatile representation for robotics. In *CVPR*, 2023. 1, 3

Shuai Bai et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. 16

Homanga Bharadhwaj, Roozbeh Mottaghi, Abhinav Gupta, and Shubham Tulsiani. Track2Act: Predicting point tracks from internet videos enables generalizable robot manipulation. In *ECCV*, 2024. 3

Homanga Bharadhwaj et al. Gen2Act: Human video generation in novel scenarios enables generalizable robot manipulation. In *CoRL*, 2025. 3

Kevin Black, Noah Brown, James Darpinian, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. In *CoRL*, 2025. 3

Samarth Brahmbhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. ContactPose: A dataset of grasps with object contact and hand pose. In *ECCV*, 2020. 3

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, et al. SAM 3: Segment anything with concepts. In *ICLR*, 2026. 16

Kaihan Chen, Yanming Shao, Haifeng Ji, Xiaokang Yang, and Yao Mu. V2P-Manip: Learning dexterous manipulation from monocular human videos. *arXiv preprint arXiv:2606.16436*, 2026. 3

Jeremy A. Collins, Cody Houff, You Liang Tan, and Charles C. Kemp. ForceSight: Text-guided mobile manipulation with visual-force goals. In *ICRA*, 2024. 3

Zichen Jeff Cui, Omar Rayyan, Haritheja Etukuru, Bowen Tan, Zavier Andrianarivo, Zicheng Teng, Yihang Zhou, Krish Mehta, Nicholas Wojno, Kevin Yuanbo Wu, Manan H. Anjaria, Ziyuan Wu, Manrong Mao, Guangxun Zhang, Binit Shah, Yejin Kim, Soumith Chintala, Lerrel Pinto, and Nur Muhammad Mahi Shafiullah. Contact-anchored policies: Contact conditioning creates strong robot utility models. In *RSS*, 2026. 3

Dima Damen et al. Scaling egocentric vision: The EPIC-KITCHENS dataset. In *ECCV*, 2018. 15

Kiana Ehsani, Shubham Tulsiani, Saurabh Gupta, Ali Farhadi, and Abhinav Gupta. Use the force, Luke! learning to predict physical forces by simulating effects. In *CVPR*, 2020. 3

Alexey Gavryushin, Xi Wang, Robert J. S. Malate, Chenyu Yang, Davide Liconti, René Zurbrügg, Robert K. Katzschmann, and Marc Pollefeys. MAPLE: Encoding dexterous robotic manipulation priors learned from egocentric videos. *arXiv preprint arXiv:2504.06084*, 2025. 3

Raghav Goyal et al. The "something something" video database for learning and evaluating visual common sense. In *ICCV*, 2017. 15

Patrick Grady, Chengcheng Tang, Christopher D. Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C. Kemp. ContactOpt: Optimizing contact to improve grasps. In *CVPR*, 2021. 3

Patrick Grady, Chengcheng Tang, Samarth Brahmbhatt, Christopher D. Twigg, Chengde Wan, James Hays, and Charles C. Kemp. PressureVision: Estimating hand pressure from a single RGB image. In *ECCV*, 2022. 3, 15

Patrick Grady, Jeremy A. Collins, Chengcheng Tang, Christopher D. Twigg, Kunal Aneja, James Hays, and Charles C. Kemp. PressureVision++: Estimating fingertip pressure from diverse RGB images. In *WACV*, 2024. 3

Kristen Grauman et al. Ego4D: Around the world in 3,000 hours of egocentric video. In *CVPR*, 2022. 1, 15

Kristen Grauman et al. Ego-Exo4D: Understanding skilled human activity from first- and third-person perspectives. In *CVPR*, 2024. 15

Jiayuan Gu, Sean Kirmani, Paul Wohlhart, Yao Lu, Montserrat Gonzalez Arenas, Kanishka Rao, Wenhao Yu, Chuyuan Fu, Keerthana Gopalakrishnan, Zhuo Xu, Priya Sundaresan, Peng Xu, Hao Su, Karol Hausman, Chelsea Finn, Quan Vuong, and Ted Xiao. RT-Trajectory: Robotic task generalization via hindsight trajectory sketches. In *International Conference on Learning Representations (ICLR)*, 2024. 3

Irmak Guzey, Haozhi Qi, Julen Urain, Changhao Wang, Jessica Yin, Krishna Bodduluri, Mike Lambeta, Lerrel Pinto, Akshara Rai, Jitendra Malik, Tingfan Wu, Akash Sharma, and Homanga Bharadhwaj. Dexterity from smart lenses: Multi-fingered robot manipulation with in-the-wild human demonstrations. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026. 2, 3

Siddhant Haldar and Lerrel Pinto. Point policy: Unifying observations and actions with key points for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2025. 2, 3

Siddhant Haldar, Lars Johannsmeier, Lerrel Pinto, Abhishek Gupta, Dieter Fox, Yashraj Narang, and Ajay Mandlekar. Point bridge: 3D representations for cross domain policy learning. *arXiv preprint arXiv:2601.16212*, 2026. 3

Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. ReKep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. In *CoRL*, 2024. 3

Siddharth Karamcheti et al. Language-driven representation learning for robotics. In *RSS*, 2023. 3

Matthieu Lepert et al. Phantom: Training robots without robots using only human videos. In *CoRL*, 2025. 3

Mara Levy, Siddhant Haldar, Lerrel Pinto, and Abhinav Shrivastava. P3-PO: Prescriptive point priors for visuo-spatial generalization of robot policies. In *ICRA*, 2025. 3

Qixiu Li, Yu Deng, Yaobo Liang, Lin Luo, Lei Zhou, Chengtang Yao, Lingqi Zeng, Zhiyuan Feng, Huizhi Liang, Sicheng Xu, Yizhong Zhang, Xi Chen, Hao Chen, Lily Sun, Dong Chen, Jiaolong Yang, and Baining Guo. Scalable vision-language-action model pretraining for robotic dexterous manipulation with real-life human activity videos. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026. 1, 2, 3, 4, 7, 15, 19

Zongmian Li, Jiri Sedlar, Justin Carpentier, Ivan Laptev, Nicolas Mansard, and Josef Sivic. Estimating 3D motion and forces of person-object interactions from monocular video. In *CVPR*, 2019. 3

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. 5

Fangchen Liu, Kuan Fang, Pieter Abbeel, and Sergey Levine. MOKA: Open-world robotic manipulation through mark-based visual prompting. In *RSS*, 2024. 3

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023. 5

Edith Llontop, Kalyan Vadrevu, and NVIDIA. NVIDIA Isaac GR00T N1.7: Open reasoning VLA model for humanoid robots. Hugging Face blog post, April 2026. URL https://huggingface.co/blog/nvidia/gr00t-n1-7. 1, 2, 3, 7, 19

Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Haiweng Xu, Qin Jin, and Zongqing Lu. Vision-language-action pretraining from large-scale human videos. In *ICML*, 2026. 3

Yecheng Jason Ma et al. VIP: Towards universal visual reward and representation via value-implicit pre-training. In *ICLR*, 2023. 3

Arjun Majumdar et al. Where are we in the search for an artificial visual cortex for embodied intelligence? In *NeurIPS*, 2023. 3

Priyanka Mandikal and Kristen Grauman. DexVIP: Learning dexterous grasping with human hand pose priors from video. In *Conference on Robot Learning (CoRL)*, 2022. 3, 19

Lucas Manuelli, Wei Gao, Peter Florence, and Russ Tedrake. kPAM: Keypoint affordances for category-level robotic manipulation. In *ISRR*, 2019. 3

Juncheng Mu, Sizhe Yang, Yiming Bao, Hojin Bae, Tianming Wei, Linning Xu, Boyi Li, Huazhe Xu, and Jiangmiao Pang. DexImit: Learning bimanual dexterous manipulation from monocular human videos. In *RSS*, 2026. 2, 3

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *CoRL*, 2022. 1, 3

Supreeth Narasimhaswamy, Trung Nguyen, and Minh Hoai. Detecting hands and recognizing physical contact in the wild. In *NeurIPS*, 2020. 3

NVIDIA. Cosmos 3: Omnimodal world models for physical AI. *arXiv preprint arXiv:2606.02800*, 2026. 15

Open X-Embodiment Collaboration. Open X-Embodiment: Robotic learning datasets and RT-X models. In *ICRA*, 2024. 1

Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9826–9836, 2024. 3

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 5

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *CVPR*, 2017. 16

Rainbow Robotics. RB-Y1: Bimanual mobile manipulator. Product documentation, 2024. URL https://rainbowrobotics.github.io/rby1-dev/. 2

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *EMNLP*, 2019. 16

Jie Ren, Zhehao Jiang, Yinhong Yang, Haorui Jia, Han Jiang, Ben Li, Yao Yao, Cheng Lin, Qiu Shen, Zhenshan Bing, Xiao-Xiao Long, and Xun Cao. C2Dex: Contact-consistent reconstruction and retargeting for dexterous manipulation from monocular video. *arXiv preprint arXiv:2608.07045*, 2026. 3

Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 36(6): 245:1–245:17, 2017. 2, 4, 15, 16

Dandan Shan, Jiaqi Geng, Michelle Shu, and David F. Fouhey. Understanding human hands in contact at internet scale. In *CVPR*, 2020. 3

Kenneth Shaw, Shikhar Bahl, and Deepak Pathak. VideoDex: Learning dexterity from internet videos. In *CoRL*, 2022. 3, 19

Anthony Simeonov, Yilun Du, Andrea Tagliasacchi, Joshua B. Tenenbaum, Alberto Rodriguez, Pulkit Agrawal, and Vincent Sitzmann. Neural descriptor fields: SE(3)-equivariant object representations for manipulation. In *ICRA*, 2022. 3

Yuxin Ray Song, Jinzhou Li, Rao Fu, Devin Murphy, Kaichen Zhou, Rishi Shiv, Yaqi Li, Haoyu Xiong, Crystal Elaine Owens, Yilun Du, et al. OpenTouch: Bringing full-hand touch to real-world interaction. *arXiv preprint arXiv:2512.16842*, 2025. 15

Tony Tao, Mohan Kumar Srirama, Jason Jingzhou Liu, Kenneth Shaw, and Deepak Pathak. DexWild: Dexterous human interactions for in-the-wild robot policies. In *Robotics: Science and Systems (RSS)*, 2025. 1

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. Vggt-ω, 2026. URL https://arxiv.org/abs/2605.15195. 16

Bowen Wen, Shaurya Dewan, and Stan Birchfield. Fast-foundationstereo: Real-time zero-shot stereo matching, 2026. URL https://arxiv.org/abs/2512.11130. 6, 16

Chuan Wen, Xingyu Lin, John So, Kai Chen, Qi Dou, Yang Gao, and Pieter Abbeel. Any-point trajectory modeling for policy learning. In *Robotics: Science and Systems (RSS)*, 2024. 3

Kevin Yuanbo Wu, Tianxing Zhou, Isaac Tu, Billy Yan, Irmak Guzey, David Fouhey, Dandan Shan, and Lerrel Pinto. Human universal grasping. *arXiv preprint arXiv:2606.17054*, 2026. 2, 3

Wuji Technology. Wuji Hand 2: 20-DoF dexterous robotic hand. Product page, 2026. URL https://wuji.tech/en/hand. 2

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 (CoRL)*, 2025. 3

Ruihan Yang, Qinxi Yu, Yecheng Wu, Rui Yan, Borui Li, An-Chieh Cheng, Xueyan Zou, Yunhao Fang, Xuxin Cheng, Ri-Zhao Qiu, Hongxu Yin, Sifei Liu, Song Han, Yao Lu, and Xiaolong Wang. EgoVLA: Learning vision-language-action models from egocentric human videos. In *IROS*, 2026a. 2, 3, 7, 19

Zihao Yang, Chengyuan Liu, Yu Zhou, Runze Lv, Tianyu Cui, Sheng Yi, Haohua Zhu, Irvine Lu, and JieQ Sun. Dexterous robot manipulation from human demonstrations via contact-anchored retargeting and residual policy learning. *arXiv preprint arXiv:2609.24093*, 2026b. 3

Chengbo Yuan, Chuan Wen, Tong Zhang, and Yang Gao. General flow as foundation affordance for scalable robot learning. In *Conference on Robot Learning (CoRL)*, 2024a. 3

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. RoboPoint: A vision-language model for spatial affordance prediction for robotics. In *CoRL*, 2024b. 3

Chi Zhang, Penglin Cai, Ziheng Xi, Haoqi Yuan, Hao Luo, Wanpeng Zhang, Sipeng Zheng, Chaoyi Xu, and Zongqing Lu. Human-centric transferable tactile pre-training for dexterous robotic manipulation. *arXiv preprint arXiv:2607.01067*, 2026a. 2, 3

Jinglei Zhang, Jiankang Deng, Chao Ma, and Rolandos Alexandros Potamias. HaWoR: World-space hand motion reconstruction from egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. 3, 15

Wenkang Zhang, Chengbo Yuan, Zicheng Zhang, Zhengxue Cheng, and Yang Gao. EgoTac: In-the-wild tactile prediction from egocentric vision. *arXiv preprint arXiv:2608.15060*, 2026b. 3

Yiming Zhao, Taein Kwon, Paul Streli, Marc Pollefeys, and Christian Holz. EgoPressure: A dataset for hand pressure and pose estimation in egocentric vision. In *CVPR*, 2025. 3

Ruijie Zheng, Dantong Niu, Yuqi Xie, Jing Wang, Mengda Xu, Yunfan Jiang, Fernando Castañeda, Fengyuan Hu, You Liang Tan, Letian Fu, Trevor Darrell, Furong Huang, Yuke Zhu, Danfei Xu, and Linxi Fan. EgoScale: Scaling dexterous manipulation with diverse egocentric human data. In *CoRL*, 2026. 3, 7, 19

Danyan Zhou, Jinxuan Lu, Jiawei Lin, Tianxing Chen, Chuqiao Lyu, and Wenbo Ding. TouchSight: Bare-handed tactile prediction from egocentric video via generative visual augmentation. *arXiv preprint arXiv:2609.20414*, 2026. 3

Xinghao Zhu, Zixi Liu, Shalin Jain, Chenran Li, Milad Noori, Michael Andres Lin, Huihua Zhao, John Welsh, Mrinal Verghese, Wei Liu, Tingwu Wang, Xingye Da, Zhengyi Luo, Vishal Kulkarni, Naema Bhatti, Yuke Zhu, Linxi Fan, Bowen Wen, Danfei Xu, Soha Pouya, and Yan Chang. Learning dexterous manipulation using contact wrench guidance from human demonstration. *arXiv preprint arXiv:2607.00033*, 2026. 3

A Hand Tracking with Contact

We use a recent hand tracker (Anonymous, 2026) that jointly estimates, from monocular egocentric video, the MANO parameters of each hand and a contact probability at each vertex of its mesh. It turns a pre-trained Cosmos 3 (NVIDIA, 2026) video diffusion backbone into a deterministic clip-level feature extractor via clean-latent conditioning, so that the backbone's priors on hand motion and hand–object interaction carry over to tracking. Its pose stream predicts anatomically plausible joint angles, *i.e.*, it disables the twist and spread of the PIP and DIP joints, and can be conditioned on the hand shape, as described next. It processes a video in clips of 81 frames, at about 7 ms of model forward time per frame for both hands on a single H100 GPU. On our demonstrations, we run it on the right view of the stereo camera with the rectified intrinsics, and it returns the 21 keypoints of each hand in the camera frame in the standard MANO layout, together with a per-frame presence flag.

Scale-consistent hand reconstruction. Crop-based per-frame trackers such as `HaWoR` (Zhang et al., 2025) re-estimate the MANO (Romero et al., 2017) shape $\beta \in \mathbb{R}^{10}$ for every frame, so the predicted hand size drifts within a single video, *i.e.*, the length of the wrist-to-middle-fingertip bone chain varies with a standard deviation of 4.6–7.4 mm on three egocentric benchmarks. They also re-estimate the metric distance of the hand independently at every frame, *e.g.*, `HaWoR` shows a depth jitter of 13.8–26.1 mm/frame^2, which directly affects the robot action space, whereas the clip-level decoding of the tracker reduces it to 1.5–1.6 mm/frame^2. To also keep the hand size consistent, the tracker adds a cached shape β_{cache} to the hand tokens of its pose stream, which keeps the hand shape consistent within a video and even across videos of the same person:

$$\mathbf{X}_H = \mathbf{q}_H + g_\beta(\beta_{\text{cache}}), \tag{4}$$

where $\mathbf{q}_H$ is a learnable query and g_β is a zero-initialized MLP. In the training phase, the ground truth β is used as β_{cache} with probability 0.5. At inference, β_{cache} is the shape predicted on the first clip in which the hand is visible in a sufficient fraction of frames, and it is reused for all clips of the video. The remaining scale–depth ambiguity is resolved at every frame against the stereo depth of the wrist (Section 3.1).

Tactile estimation. A separate tactile expert stream, trained with the pose stream frozen, predicts contact over the hand surface. It keeps one token per bone of the MANO skeleton and spreads it to the mesh vertices with the fixed linear blend skinning (LBS) weights of MANO, which avoids a costly attention over the vertices. Its output is a contact probability at each vertex, together with a force that we do not use. The contact output is trained with a positively re-weighted binary cross-entropy on per-vertex labels, *i.e.*, pressure above 1 kPa measured by a tactile glove (Song et al., 2025) and a pressure pad (Grady et al., 2022), and a signed distance of at most 5 mm to the object surface in hand–object datasets with object meshes.

Contact pooling. The tracker outputs a contact probability at each of the 778 vertices of the MANO mesh. We aggregate these probabilities at the 21 keypoints by a weighted sum with a fixed vertex-to-keypoint weight map, *i.e.*, the skinning weights of the MANO template with the fingertip regions separated from the distal joints, normalized to sum to one over the keypoint. The result lies in $[0, 1]$ and grows with the area of the hand surface in contact around the keypoint, so we interpret it as how firmly the keypoint presses, *i.e.*, its contact intensity.

B Data Pipeline Details

Human videos. The VITRA corpus (Li et al., 2026) aggregates four egocentric human-video collections, *i.e.*, `Ego4D` (Grauman et al., 2022), `Ego-Exo4D` (Grauman et al., 2024), `Something-Something v2` (Goyal et al., 2017), and `EPIC-KITCHENS` (Damen et al., 2018), with about 1M episodes of video in total. It provides per-frame language captions, for which we use the pre-processed captions of its open-source repository, and MANO hand tracks reconstructed with `HaWoR` (Zhang et al., 2025). Our demonstrations are recorded per task with an ego-view ZED Mini stereo camera and the operator's bare hands. Since no robot is involved, we collect our 200 demonstrations in about 0.5 person-hours, *i.e.*, 100 for Pick & Place (25 per object) and 50 for each bimanual task (Appendix F).

Object points. For our demonstrations, each task specifies the names of its objects, *e.g.*, "gray ball" and "white container". For the pretraining corpus, Qwen3-VL-8B-Instruct (Bai et al., 2025) reads the first frame and the caption of each episode and returns the task-relevant objects, *e.g.*, {bottle, bowl} for "pick the bottle and place it on the bowl". In both cases, we use each name as a text prompt for SAM 3.1 (Carion et al., 2026), which segments the object in the first frame and tracks it through the video with its memory, and we lift 512 points sampled uniformly at random from each mask to 3D, after discarding mask pixels on depth edges or outside 0.2–3 m. For our demonstrations, the depth comes from the stereo pair, computed by a Fast-FoundationStereo (Wen et al., 2026). For the pretraining corpus, only single-view videos are available, so we estimate the relative depth of the scene with VGGT-Omega (Wang et al., 2026) and scale it so that it agrees with the wrist depth of the HaWoR hand tracks. Camera intrinsics come from each source dataset's metadata for the pretraining corpus and from the stereo calibration for our demonstrations. In both cases, we express the points of a sample in the camera frame at its observation time.

Hand keypoints. We use the standard 21-joint MANO layout (Romero et al., 2017), *i.e.*, the 16 joints of the MANO joint regressor and the five fingertips taken from fixed mesh vertices (Appendix A). For the pretraining corpus, we read the 21 joints off the posed meshes of VITRA's HaWoR tracks, since coarse labels suffice to learn general hand–object motion. For our demonstrations, whose keypoints define the tasks, we use the hand tracker (Appendix A) and resolve the remaining scale-depth ambiguity between the hand and the stereo-measured object points at every frame. Specifically, we read the depth of the wrist surface from the stereo depth map, add 2 cm to approximate the wrist center, and scale each hand about the camera center so that its wrist lies at this depth, which preserves its projection in the image. Then, we smooth the per-frame scale with a temporal median filter and interpolate it over frames in which the wrist is occluded. The teleoperated demonstrations pass through the same object pipeline, and their hand keypoints come from forward kinematics of the measured joint states (Appendix E).

C Architecture Details

Observation encoding. Each observation stream is encoded into 512-dim tokens, and each group of tokens is preceded by a learned delimiter token, *i.e.*, one each for the language, the objects, the camera, and the hands. The language instruction is encoded by a Sentence Transformer (Reimers & Gurevych, 2019) and projected by an MLP into a single token. Each task-relevant object contributes a *semantic* token, obtained by encoding its name with the same Sentence Transformer and an MLP, followed by a *geometry* token, obtained by encoding its 512 points with PointNet (Qi et al., 2017), *i.e.*, a shared per-point MLP followed by max-pooling. We cap the number of objects at four and fill unused slots with a zero embedding. The 21 keypoints of each hand are concatenated over the two hands into a 126-dim vector and projected by an MLP into one hand token, where an absent hand is zero-padded. Finally, the camera extrinsics are projected by an MLP into one more token. Since every point of a sample is expressed in the camera frame at its observation time, the extrinsics are the identity throughout our data.

Diffusion transformer. The DiT consists of 12 blocks with adaptive layer normalization, a width of 768, and 12 attention heads, and a linear layer maps the observation tokens to this width. The $H = 16$ action tokens cover 0.8 s at 20 Hz. The sequence is processed with causal self-attention, so that every action token attends to all observation tokens and to the action tokens before it, and the flow time is the only input to the adaptive layer normalization. The target chunk is normalized per step and coordinate with statistics of the training set. At inference, we integrate the velocity from noise with 6 Euler steps, add the resulting displacements to the current keypoints, and read the contact intensities from the last step. When a hand is absent, we set its noise to zero, exclude its keypoints and contact from both losses through per-hand masks, and ignore its predictions at deployment. We treat the unused hand of our unimanual task in the same way.

Reference frame. The camera is mounted to approximate the egocentric viewpoint of the human demonstrations and stays fixed during our tasks, so the camera frame at deployment is also the world frame. We note that this is a convention rather than a constraint. Specifically, the policy consumes and emits 3D points, and the camera pose relative to the robot base follows from forward kinematics, so a

different camera pose amounts to a known rigid transform into the training convention and requires no retraining. Nevertheless, we have not evaluated viewpoint changes experimentally.

D Training Details

Pretraining. We pretrain on the VITRA corpus ($\sim$1M egocentric episodes) for 100k optimizer steps on four NVIDIA H100 GPUs with a batch size of 256 per GPU, using AdamW with a learning rate of 10^{-4} and a weight decay of 10^{-4}. Training uses bf16 mixed precision, gradient clipping at $\|g\| \leq 1$, and a linear warmup over the first 1k steps followed by a constant learning rate. During pretraining only, we add Gaussian noise with a standard deviation of 1 cm to the object points. Each forward pass generates an action chunk of $H = 16$ steps (Section 3.2), and pretraining takes approximately 6h.

Fine-tuning. We fine-tune from the pretrained checkpoint for 60k steps with a batch size of 128 and without object-point noise, keeping the optimizer, learning rate, weight decay, gradient clipping, and mixed precision of pretraining. The contact weight in Equation (2) is $\lambda = 1$. Fine-tuning a single policy for all three tasks takes approximately 2h on a single A100 GPU. We apply the flow-matching loss to the 42 hand keypoints and the contact loss to their intensities, while the object-point tokens carry observations and receive no supervision. We zero-pad a hand absent from a clip, as well as the unused hand of the unimanual task, and mask it out of both losses. When teleoperated demonstrations are included, they are pooled with the human demonstrations of all three tasks, and each training sample is a frame drawn uniformly at random from an episode drawn uniformly at random from the pool, *i.e.*, without a fixed ratio between human and robot data.

Ablation variants. The *w/o contact labels* variant of Table 3 sets $\lambda = 0$ during pretraining and keeps the fine-tuning unchanged, so its contact head learns only from the contact labels of our human demonstrations. The *binary labels* variant thresholds the contact intensities at 0.2 in both the pretraining corpus and our human demonstrations, close to the Otsu threshold of their distribution on our demonstrations (0.18), and is trained with the same loss.

E Robot Hand Setup

Keypoint sites and normals. The kinematic model of each hand carries 21 sites at the locations corresponding to the MANO keypoints. Since the Wuji hand follows the joint layout of the human hand, *i.e.*, four joints per finger including the thumb's CMC, we map the keypoints one-to-one onto its model without hand-crafted offsets or manually placed sites: the wrist, CMC, MCP, PIP, IP, and DIP keypoints are the anchors of the corresponding joints, the fingertips are the tip sites of the manufacturer's model, and the wrist is the origin of the wrist body. For contact-force injection (Section 3.3), we estimate the normal at each site from the flexion direction of the motor that drives it, *i.e.*, the direction in which the site moves when that motor closes the finger, and we recompute it at every pose rather than fixing it at a reference pose.

Inverse kinematics. Since human and robot hands share the 21-keypoint layout, retargeting reduces to inverse kinematics on the displaced targets $\tilde{\mathbf{x}}_{1:21}$ of each hand (Equation (3)). First, we rebuild the predicted hand with the robot hand's own bone lengths while keeping the predicted bone directions, since the robot cannot shorten a link to match a smaller predicted hand and would otherwise curl its fingers instead. Then, we solve the IK in two stages per side, with the torso and the head fixed. The seven arm joints first track the wrist and the four finger MCP keypoints, *i.e.*, the palm, and the 20 joints of the hand then track all keypoints with the arm fixed. Each stage minimizes a weighted sum of squared keypoint errors with a small temporal regularization toward the previous solution, subject to the joint limits, using L-BFGS-B. The weight of each keypoint grows with its predicted contact, *i.e.*, $w_k \propto b_k(1 + \gamma\,\hat{c}_k)$ with a base weight b_k and $\gamma = 2$. The contact-driven displacement enters only the hand stage, *i.e.*, we solve it once for the predicted targets, evaluate the normals at that pose, and solve it again for the displaced targets. With 21 targets per hand, the hand stage is over-determined on the 20-DoF Wuji hand, so the robot hand cannot reach every target exactly, and the least-squares solution distributes the residual over the sites.

Wuji hands on the RB-Y1. The Wuji Hand 2 is an anthropomorphic hand with 20 actuated joints, four per finger, and two of them are mounted on the seven-DoF arms of the RB-Y1. A ZED Mini stereo camera mounted on the robot's head provides the observation, and we express all points in its right rectified camera frame, whose pose follows from the calibrated extrinsics and the head and torso joints, which stay fixed during our tasks. The joint targets of each 20 Hz step are linearly interpolated to 100 Hz for the arms and 500 Hz for the hands, and each step is limited by the joint velocity limits of the robot's safety configuration. After executing a chunk, the robot holds its last command for two steps, *i.e.*, 100 ms, before the next observation, so that the arms settle on their targets. We fine-tune the policy on the RB-Y1's own teleoperated demonstrations (Appendix H).

F Task Setup

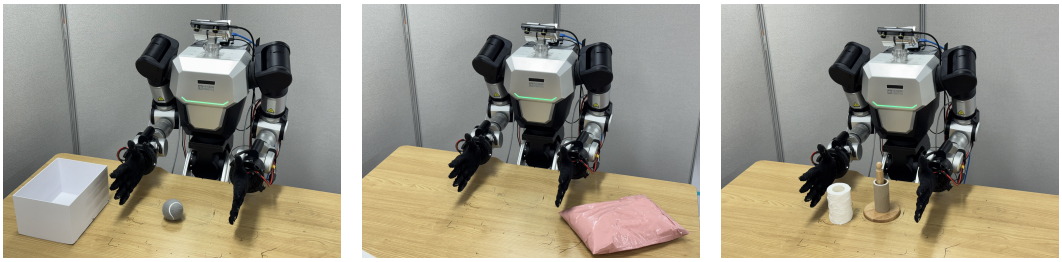


Figure 5: **Task setups.** Left: Pick & Place (unimanual). Middle: Flip Plastic Bag (bimanual). Right: Replace Tissue Roll (bimanual).

Pick & Place (unimanual). The objects are a bottle, a box, a doll, and a ball, and the same policy handles all four. Each object is evaluated once at each position on a 4×4 grid. The target container is placed at a fixed position to the right of the robot. The robot uses its right hand to pick up the object and place it in the container. Total of 100 human demonstrations were collected, with 25 demonstrations per object.

Flip Plastic Bag (bimanual). We use a 25×30 cm plastic bag, randomly initialized on the left side of the robot. The robot picks up the bag with its left hand and flips it over to reveal the label. It then uses its right hand to push the flipped bag toward the target region on its right. Total of 50 human demonstrations were collected for this task.

Replace Tissue Roll (bimanual). The tissue roll is cylindrical, measuring 3×4 cm (diameter $\times$ height), while the empty core measures 2×4 cm. The tissue holder is 7 cm tall and is placed between the robot's hands, initially holding the empty core. The robot removes the empty core with its left hand and places a new roll onto the holder with its right hand. The new roll is randomly initialized within a circular region of 5 cm diameter to the right of the holder. Total of 50 human demos were collected for this task.

G Success Criteria

We define task success as follows. Success judgments are made by a human evaluator observing the trial in real time. We run 16 trials per task and object, *i.e.*, 64 trials for Pick & Place with its four objects and 16 trials for each bimanual task. On the bimanual tasks, a trial that completes only one of the two subtasks counts as a partial success and scores 0.5.

- **Pick & Place.** A trial is successful if the robot grasps the target object, lifts and transports it to the container, and releases it, placing the object inside the container. Timeout: 30 seconds.
- **Flip Plastic Bag.** A trial is successful if the bag is flipped over and pushed into the target region. A trial is partially successful if the bag is flipped over but does not reach the target region. Timeout: 1 minute.

Table 4: **Teleoperated demonstration pools.** Number of teleoperated demonstrations M collected for each task on the RB-Y1 with Wuji hands; the 10%, 50%, and 100% regimes of Table 1 use a tenth, a half, and all of the pool, respectively.

Robot	Pick & Place	Flip Plastic Bag	Replace Tissue Roll
RB-Y1 + Wuji	100	50	50

- **Replace Tissue Roll.** A trial is successful if the empty core is removed from the holder and the new roll is placed onto the holder. A trial is partially successful if the empty core is removed but the new roll is not successfully placed. Timeout: 1 minute.

H Teleoperated Demonstrations

Expert operators collect 200 teleoperated demonstrations with a pair of MANUS gloves and two VIVE trackers in about 3 person-hours, *i.e.*, a pool of $M = 100$ for Pick & Place (25 per object) and $M = 50$ for each bimanual task (Table 4). The 10% and 50% regimes keep a tenth and a half of each pool, *i.e.*, 10 and 50 demonstrations for Pick & Place and 5 and 25 for each bimanual task, and the 100% regime the whole pool, with the same subsets for every method. Teleoperated demonstrations enter each method in its native action space, *i.e.*, joint trajectories for VITRA, MANO parameters for EgoVLA, relative end-effector actions for GR00T N1.7, and, for ConAct, the 21 keypoints of each hand from forward kinematics of the measured joint states, without contact labels, so that the contact loss is masked on them (Section 3.2).

I Baseline Details

VITRA. We use the VITRA model (Li et al., 2026), which is pretrained on the same egocentric video corpus that ConAct uses for pretraining. Since VITRA's hand action space is defined in robot joint space, we construct the fine-tuning supervision from the human demonstrations as follows. For each demonstration, we run our hand-point extraction pipeline to obtain the 21-keypoint trajectory of each hand, and then convert it to a robot joint trajectory using the same IK solver used at ConAct's deployment (Section 3.3). Teleoperated demonstrations are used in their native joint space. At deployment, VITRA's predicted joint angles are sent directly to the controller, *i.e.*, bypassing the IK stage that ConAct uses to map keypoint targets to joint commands. We fine-tune it for 60k steps with a global batch size of 64.

EgoVLA. EgoVLA (Yang et al., 2026a) pretrains a Vision-Language-Action model on egocentric human video with hand pose as an intermediate action. We fine-tune it from its pretrained checkpoint on the same human demonstrations as ConAct. At deployment, EgoVLA converts the predicted wrist pose to robot arm joints via inverse kinematics and maps the predicted hand pose to robot hand joints using its retargeting MLP. We train its retargeting MLP once using all 200 robot demonstrations and kept fixed upon all data regimes, to ensure consistent deployment on our evaluation setup.

GR00T N1.7. GR00T N1.7 (Llontop et al., 2026) is pretrained on cross-embodiment robot data and on the 20K hours of egocentric human video of EgoScale (Zheng et al., 2026) with a relative end-effector action space shared between human and robot. Following EgoScale (Zheng et al., 2026), we convert our human demonstrations into corresponding robot-compatible action representations. We fine-tune GR00T N1.7 from its public checkpoint for 60k steps with a global batch size of 64, using our converted human and teleoperated robot demonstrations.

Other methods. DexVIP (Mandikal & Grauman, 2022) and VideoDex (Shaw et al., 2022) also learn multi-fingered policies from human videos, but they have no public implementation, so we do not compare against them.

J Potential Societal Impact

ConAct lowers the number of teleoperated demonstrations needed to deploy multi-fingered hands, which can make dexterous robots more accessible, *e.g.*, for household assistance and manufacturing. However, it also raises two risks. First, the force that ConAct applies is set by contact intensities estimated from videos rather than measured, so a policy can press harder than intended on fragile objects or on people nearby. Hence, we recommend deploying it with joint torque or current limits and under human supervision, especially in assistive settings. Second, human videos can capture bystanders and private spaces. Our pretraining corpus is built from publicly released egocentric datasets (Section 3.1), and we record the in-house demonstrations with the consent of the operators.