ROBOTRYON: SYNTHESIZING DEXTEROUS ROBOT DEMONSTRATIONS FROM HUMAN VIDEOS VIA VIRTUAL TRY-ON

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot foundation models have shown remarkable performance in robot control, yet adapting them to a new task still relies heavily on teleoperated demonstrations. Egocentric human videos offer a scalable substitute, but retargeting human poses into robot actions or overlaying rendered robots on human leaves significant visual or action gaps for dexterous hands with high-DoF. To address this limitation, we propose RoboTryOn, which treats human-to-robot translation as virtual try-on: try-on keeps the person and swaps the garment, RoboTryOn keeps the demonstrated motion and swaps the human for the robot. Specifically, RoboTryOn synthesizes robot data using masks and hand skeletons as conditioning signals, while reference images specify the target robot embodiment. To maintain visual consistency, we translate selected keyframes autoregressively and use them to guide video synthesis conditioned on the skeleton. Additionally, we use an inverse dynamics model (IDM) that incorporates depth features to infer action labels from synthesized videos for policy learning. Our experiments across multiple humanoid embodiments show that policies trained with our synthesized data improve the average success rate on novel tasks, including challenging bimanual manipulation, from 2% to 60%, without any real robot demonstration of those tasks. Moreover, training with our synthetic data matches the performance of policies trained solely on teleoperated demonstrations at approximately a third of the collection cost, demonstrating the cost efficiency of our approach.

1 INTRODUCTION

Robot foundation models trained on large-scale demonstrations have made substantial progress toward general-purpose robot control (Open X-Embodiment Collaboration, 2024; Octo Model Team et al., 2024; Kim et al., 2024; Black et al., 2024; Liu et al., 2025a; NVIDIA et al., 2025; Black et al., 2025), while extending them to new tasks and environments still requires new robot demonstrations of those tasks and environments (Open X-Embodiment Collaboration, 2024; Khazatsky et al., 2024; Bu et al., 2025). Collecting such demonstrations through teleoperation requires robot hardware, a teleoperation system, and substantial operator effort for every episode (Zhao et al., 2023; Wu et al., 2024; Chi et al., 2024; Khazatsky et al., 2024). The burden is heaviest for dexterous hands, where the operator must coordinate many degrees of freedom and fine-grained contacts through a retargeting interface that is itself difficult to build and to use (Handa et al., 2020; Qin et al., 2023; Cheng et al., 2024; Wang et al., 2024; Xu et al., 2025).

Human videos offer another way to demonstrate a new task. Egocentric recordings of human object manipulation already exist at internet scale, from in-the-wild collections (Goyal et al., 2017; Shan et al., 2020; Damen et al., 2022; Grauman et al., 2022; 2024) to manually annotated corpora recorded for robot learning (Liu et al., 2022; Hoque et al., 2025; Li et al., 2025; Punamiya et al., 2026), and provide far more demonstrations than teleoperation can produce. They capture how people grasp and manipulate objects, including coordinated bimanual interaction, without a robot during collection. Recent works use these videos to learn transferable visual and behavioral knowledge (Nair et al., 2022; Yang et al., 2025; Luo et al., 2025; Dyna Robotics, 2026), or convert them into robot data through retargeting (Kareer et al., 2025), rendering, (Lepert et al., 2025b;a; Wang et al., 2026; Liu et al., 2026), or generative translation (Song et al., 2025; Ci et al., 2025).

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.

However, a human video is separated from a robot demonstration by a visual gap and an action gap, and both are severe for dexterous hands (Liu et al., 2025b; Punamiya et al., 2025; Liu et al., 2026). Visually, the frames show a human arm and hand where the policy will see a robot. Prior work narrows this gap by masking the human arm or overlaying a rendered robot, and must apply the same edit to the robot's camera stream during both training and deployment (Kareer et al., 2025; Lepert et al., 2025b;a). In action, the reconstructed hand pose is itself imprecise, and no exact retargeting map takes it to robot joint commands, since human and robot hands differ in morphology and kinematics (Handa et al., 2020; Qin et al., 2023; Kong et al., 2026); even a perfect map yields a kinematic state rather than a control command. The gap is widest during contact. When a hand holds an object, for example, it squeezes harder than the video shows (Chen et al., 2025; Liu et al., 2026).

Our approach. We address these gaps by treating human-to-robot transfer as virtual try-on, which keeps the motion and changes the embodiment. RoboTryOn converts a human manipulation video into a paired robot demonstration with action labels. To this end, we first generate a target robot video performing the same task in the source scene and then predict the corresponding joint-space actions with an inverse dynamics model (IDM), rather than deriving robot actions directly from reconstructed hand motion. As in keyframe-based video virtual try-on (Zuo et al., 2025), we autoregressively translate selected human frames into robot keyframes, and synthesize the motion between consecutive keyframes under the guidance of the human hand skeleton flow. Our generative models learn this translation from robot videos through a shared arm mask and hand skeleton interface, allowing the same conditions to be extracted from human videos at inference without paired human and robot demonstrations. We then train a policy on the synthesized robot demonstrations, enabling the robot to learn new tasks with few or no teleoperated demonstrations.

Our experiments show that RoboTryOn consistently improves policy performance when robot demonstrations of a task are limited or unavailable. In particular, averaged over two robot embodiments, RoboTryOn increases success from 2.1% to 60.0% on out-of-domain tasks, *i.e.*, tasks involving goals, skills, and objects unseen in real robot demonstrations. It also achieves 45.9% on challenging bimanual tasks despite all teleoperated demonstrations containing only single-arm manipulation. Finally, training with demonstrations derived from human video matches the performance of training on teleoperation alone while requiring approximately a third of the collection cost (see Figure 5).

We summary our main contributions below:

- We introduce RoboTryOn, which turns an egocentric human video into a robot demonstration the way virtual try-on swaps a garment, with generative models trained on robot data alone.
- We show that our synthesized demonstrations bridge the visual and action gaps in human videos, removing the need for test-time visual editing and generating better action labels than retargeted actions.
- Through experiments on two real robots with dexterous hands over diverse out-of-domain tasks, we demonstrate that our synthesized demonstrations enable learning tasks without teleoperated demonstrations and reduce data collection costs relative to teleoperation.

2 Related Work

Learning from human videos. Human videos provide a scalable source of visual and behavioral knowledge for robot learning. Early work learns visual representations from human videos for downstream robot control (Nair et al., 2022; Majumdar et al., 2023), while recent approaches incorporate human videos into robot foundation model training to learn transferable visual and behavioral knowledge (Yang et al., 2025; Luo et al., 2025; Dyna Robotics, 2026; NVIDIA et al., 2025). Another line converts human demonstrations into robot training data through motion retargeting (Kareer et al., 2025), often combined with rendered robot observations (Lepert et al., 2025b;a; Wang et al., 2026). Approaches based on simulation further construct digital twins to generate robot observations and refine action trajectories (Liu et al., 2026).

Generative human-to-robot translation. Recent advances in video diffusion models have enabled direct translation of human demonstrations into robot videos. Mitty (Song et al., 2025) conditions generation on human videos and learns from paired human–robot data, including synthetically ren-

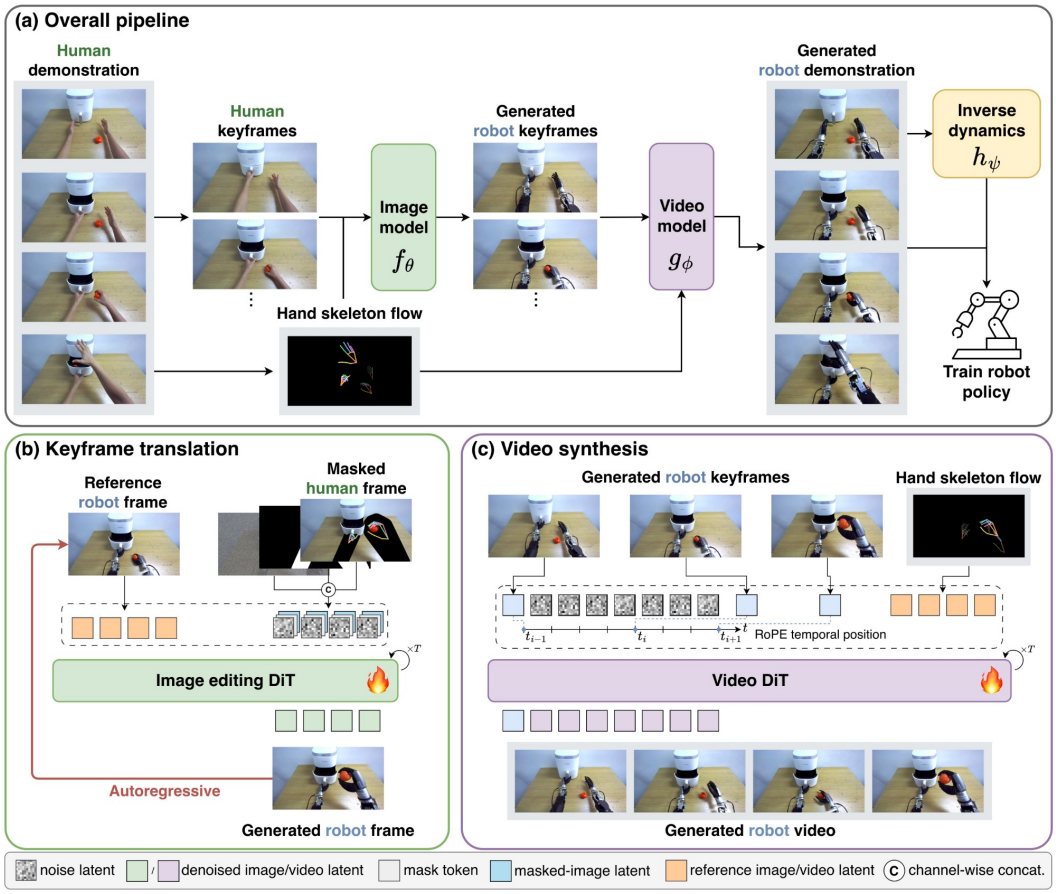


Figure 1: **RoboTryOn pipeline.** (a) The image model sequentially translates keyframes of a human video into robot keyframes, the video model synthesizes the motion between them under the guidance of the hand skeleton flow, and the inverse dynamics model labels the actions. (b) The masked human frame is inpainted with the previous robot keyframe as a reference. (c) Robot keyframes at their temporal positions and the skeleton flow condition the video model.

dered pairs. H2R-Grounder (Ci et al., 2025) removes the need for paired supervision by reconstructing robot videos from inpainted backgrounds and gripper pose cues. Our approach supports multiple dexterous embodiments through shared generative models and produces demonstrations with action labels for real-world policy learning.

Video synthesis and action labeling. Constructing robot demonstrations through video generation requires both temporally consistent synthesis and corresponding action labels. Video virtual try-on studies consistency when modifying a subject's appearance (Fang et al., 2024; Zuo et al., 2025); in particular, DreamVVT generates keyframes and uses them to guide video synthesis. We adopt this decomposition for embodiment translation. To obtain action supervision from the resulting videos, inverse dynamics models estimate actions from observed state transitions. Prior work uses such models to label unlabeled videos (Baker et al., 2022) or recover actions from generated robot videos (Du et al., 2023; Jang et al., 2025).

3 Method

3.1 Problem Setup and Overview

Policy learning from demonstrations. We consider a teleoperated robot demonstration dataset $\mathcal{D} = \{\mathbf{X}_i\}_{i=1}^{N}$, where each demonstration $\mathbf{X}_i = [\{\mathbf{o}_t, \mathbf{q}_t, \mathbf{a}_t\}_{t=1}^{T_i}, \mathbf{c}]$ contains a visual observation $\mathbf{o}_t$, a proprioceptive state $\mathbf{q}_t$, and a robot action $\mathbf{a}_t$ at each timestep $t \in [1, T_i]$, together with a task instruction $\mathbf{c}$. We mainly use vision-language-action models (VLAs) as our base policy π_ω, which predicts the next action chunk $\mathbf{A}_t = \mathbf{a}_{t:t+H-1}$ of horizon H conditioned on $(\mathbf{o}_t, \mathbf{q}_t, \mathbf{c})$. Following

recent VLAs (Black et al., 2024; 2025; NVIDIA et al., 2025), we fine-tune the pre-trained VLA with the flow-matching objective (Lipman et al., 2023; Liu et al., 2023) to denoise the noisy future action chunk with diffusion timestep $\tau \in [0, 1]$, *i.e.*, $\pi_\omega(\mathbf{A}_t^\tau | \mathbf{o}_t, \mathbf{q}_t, \mathbf{c})$.

Demonstrations from human video. RoboTryOn aims to augment $\mathcal{D}$ with demonstrations synthesized from human video to address the expensive cost of collecting real robot data. A human demonstration $\mathbf{X}^h = [\{\mathbf{o}_t^h\}_{t=1}^T, \mathbf{c}]$[1] contains the observations with egocentric view $\mathbf{o}_t^h$ and the instruction $\mathbf{c}$. Given $\mathbf{X}^h$ and a reference image of the target robot r, RoboTryOn uses a three-stage pipeline to generate a robot demonstration $\hat{\mathbf{X}}^r = [\{\hat{\mathbf{o}}_t^r, \hat{\mathbf{q}}_t, \hat{\mathbf{a}}_t\}_{t=1}^T, \mathbf{c}]$ that follows the human motion (see Figure 1). First, an image generation model f_θ translates selected human keyframes into robot keyframes (Section 3.2). Next, a video generation model g_ϕ synthesizes the robot video between keyframes under the guidance of the hand skeleton (Section 3.3). Finally, an inverse dynamics model h_ψ predicts the action labels $\hat{\mathbf{a}}$ from the synthesized video, and the states $\hat{\mathbf{q}}$ are then derived from the predicted actions (Section 3.4). As the results, the synthesized demonstrations form $\hat{\mathcal{D}}$, and train the policy π_ω on either $\hat{\mathcal{D}}$ alone or a mixture of $\hat{\mathcal{D}}$ and $\mathcal{D}$.

3.2 Keyframe Translation

In this section, we describe how RoboTryOn select keyframes from the human demonstration $\mathbf{X}^h$ and translate them into target robot keyframes. We first train the image generation model f_θ on robot demonstration $\mathbf{X}$ and apply it to the selected human keyframes at inference.

Keyframes. We begin by selecting keyframes from the human wrist trajectories (Li et al., 2025). We select a frame when both wrists are moving slowly and at least one wrist has moved a minimum distance from its position in the last selected keyframe, and we always select the first and last frames (details in Section B.2.1). We denote $\mathcal{T} = \{t_1 < \cdots < t_L\}$ as the selected keyframe indices.

Conditioning. The image generation model f_θ is conditioned on an arm and hand mask $\mathbf{M}_t$ and a hand skeleton $\mathbf{S}_t$. Following recent virtual try-on works (Choi et al., 2021; 2024; Jiang et al., 2024; Lee & Kwak, 2025), $\mathbf{M}_t$ is enlarged and smoothed so that its size and outline do not reveal the embodiment it covers. Specifically, $\mathbf{S}_t$ is the 2D projection of a common set of hand keypoints. For robot frames, we obtain these keypoints through forward kinematics. For human frames, we reconstruct the hand (Zhang et al., 2025), retarget it to the target robot hand, and project the resulting keypoints. Since the retargeting used at inference is imprecise, we make the model robust to it during training by perturbing and randomly dropping the keypoints obtained from forward kinematics (see Section B.2.2). The model thus learns to use $\mathbf{S}_t$ as a coarse guide to hand position and motion and to infer the detailed hand configuration from its learned prior over robot hands. We denote $\tilde{\mathbf{o}}_t$ for the robot frame $\mathbf{o}_t$ with the region $\mathbf{M}_t$ masked out and the skeleton $\mathbf{S}_t$ drawn over it. Likewise, $\tilde{\mathbf{o}}_t^h$ denotes the human frame $\mathbf{o}_t^h$ processed in the same way.

Model. The image model f_θ aims to translate a human keyframe into a robot keyframe as inpainting guided by a reference image: it fills the masked region $\mathbf{M}_t$ with the target robot in the pose specified by the skeleton $\mathbf{S}_t$, using the reference image to determine its appearance (see Figure 1(b)).

We train f_θ to denoise a robot frame $\mathbf{o}_t$ from two conditions. The first is the conditioned frame $\tilde{\mathbf{o}}_t$ defined above, together with its mask $\mathbf{M}_t$. The second is the reference frame $\mathbf{o}_{t-\delta}$, a frame from the same episode δ steps earlier. With a pretrained image encoder $\mathcal{E}_\mathrm{i}$, let $\mathbf{z} = \mathcal{E}_\mathrm{i}(\mathbf{o}_t)$, $\tilde{\mathbf{z}} = \mathcal{E}_\mathrm{i}(\tilde{\mathbf{o}}_t)$, and $\mathbf{z}_\mathrm{ref} = \mathcal{E}_\mathrm{i}(\mathbf{o}_{t-\delta})$, and let $\mathbf{m}_t$ be $\mathbf{M}_t$ resized to the latent resolution. We concatenate the conditioned frame's latent $\tilde{\mathbf{z}}$ and the mask $\mathbf{m}_t$ channel-wise with the noisy target latent, so that they are spatially aligned with the target. The reference latent $\mathbf{z}_\mathrm{ref}$ is appended as in-context tokens, from which the model copies the embodiment, the objects, and the background. The two stereo views are generated as one token sequence (details in Section B.2.3). We train f_θ using the standard flow-matching objective with the noise scale τ (see details in Section E):

$$\mathcal{L}_\mathrm{img}(\theta) = \mathbb{E}_{(\mathbf{o}_t,\, \mathbf{o}_{t-\delta})\sim\mathcal{D},\, \epsilon,\, \tau} \left[\left\| f_\theta\big([\mathbf{z}^\tau;\, \tilde{\mathbf{z}};\, \mathbf{m}_t],\, \tau;\, \mathbf{z}_\mathrm{ref}\big) - (\mathbf{z} - \epsilon) \right\|_2^2 \right], \tag{1}$$

where $[\,\cdot\,;\,\cdot\,]$ denotes channel-wise concatenation. At inference, the image model f_θ autoregressively generates the target robot keyframe $\hat{\mathbf{o}}_t^r$ from the conditioned human frame $\tilde{\mathbf{o}}_t^h$, its mask $\mathbf{M}_t$, and the

[1] For simplicity, we omit the demonstration index.

reference frame $\hat{\mathbf{o}}^r_{t^-}$, *i.e.*, the previously translated robot keyframe:

$$\hat{\mathbf{o}}^r_t \sim f_\theta\big(\cdot \mid \tilde{\mathbf{o}}^h_t,\ \mathbf{M}_t,\ \hat{\mathbf{o}}^r_{t^-}\big), \qquad t \in \mathcal{T}, \tag{2}$$

where $t^- \in \mathcal{T}$ is the keyframe time immediately preceding t. For the first keyframe, $\hat{\mathbf{o}}^r_{t^-}$ is a reference image of the target robot, which selects the embodiment (see details in Section B.2.3).

3.3 Video Synthesis

The video generation model g_ϕ synthesizes the robot video between the translated robot keyframes $\hat{\mathbf{o}}^r_{\mathcal{T}}$ (see Figure 1(c)). At the train time, g_ϕ is conditioned on set of robot keyframes $\mathbf{o}_{\mathcal{T}}$ and the hand skeleton flow $\mathbf{S}_{1:T}$. We train g_ϕ to denoise the robot video latent $\mathbf{w} = \mathcal{E}_{\mathrm{v}}(\mathbf{o}_{1:T})$, where $\mathcal{E}_{\mathrm{v}}$ is a pretrained video encoder, using the standard flow-matching objective with the noise scale τ (see details in Section E). The training objective is as follows:

$$\mathcal{L}_{\mathrm{vid}}(\phi) = \mathbb{E}_{\mathbf{o}_{1:T}\sim\mathcal{D},\ \epsilon,\ \tau}\left[\left\|g_\phi\big(\mathbf{w}^\tau, \tau;\ \mathbf{o}_{\mathcal{T}},\ \mathbf{S}_{1:T}\big) - (\mathbf{w} - \epsilon)\right\|_2^2\right]. \tag{3}$$

At inference, g_ϕ synthesizes the robot video $\hat{\mathbf{o}}^r_{1:T}$ conditioned on $(\hat{\mathbf{o}}^r_{\mathcal{T}}, \mathbf{S}_{1:T})$, as follows:

$$\hat{\mathbf{o}}^r_{1:T} \sim g_\phi\big(\cdot \mid \hat{\mathbf{o}}^r_{\mathcal{T}},\ \mathbf{S}_{1:T}\big). \tag{4}$$

A long video is synthesized in clips cut at keyframe times, each starting from the last frame of the previous clip (see details in Section B.2.4).

3.4 Pseudo Action Labeling

Finally, we use the inverse dynamics model h_ψ to infer robot action labels $\hat{\mathbf{a}}_{1:T}$ from the synthesized robot video $\hat{\mathbf{o}}^r_{1:T}$.

From a pair of stereo observations K steps apart, it predicts the chunk of joint-space actions between them. We train h_ψ on robot episodes with the recorded actions as targets, using the standard flow-matching objective with the noise scale τ (see details in Section E):

$$\mathcal{L}_{\mathrm{idm}}(\psi) = \mathbb{E}_{(\mathbf{o}_t,\, \mathbf{o}_{t+K},\, \mathbf{a}_{t:t+K})\sim\mathcal{D},\ \epsilon,\ \tau}\left[\left\|h_\psi\big(\mathbf{a}^\tau_{t:t+K}, \tau;\ \mathbf{o}_t,\ \mathbf{o}_{t+K}\big) - (\mathbf{a}_{t:t+K} - \epsilon)\right\|_2^2\right]. \tag{5}$$

At inference, h_ψ labels the synthesized video as $\hat{\mathbf{a}}_{t:t+K} \sim h_\psi(\cdot \mid \hat{\mathbf{o}}^r_t, \hat{\mathbf{o}}^r_{t+K})$ over the whole sequence, and the states $\hat{\mathbf{q}}_t$ are derived from the action labels (see details in Section B.2.6).

h_ψ takes stereo observations and builds on the features of a pretrained depth estimation model (Section 4.1). It recovers the hand motion well, whether the hand closes on an object or releases it, but the wrist position less so, since a small error in the arm joints shifts the whole hand. The hand reconstruction has the opposite profile. As Section 4.3 shows, it is too imprecise for the grasp, but it places the wrist reliably, and the stereo views refine that position further (Section B.2.2). We therefore correct the wrist position of the action labels $\hat{\mathbf{a}}_{1:T}$ with the reconstructed wrist position, while the hand configuration is kept as predicted by the IDM (Section B.2.6). Section 4.4 ablates the correction. This completes a synthesized demonstration $\hat{\mathbf{X}}^r$ of $\hat{\mathcal{D}}$, which trains the policy in the same way as a teleoperated one.

4 Experiments

We validate RoboTryOn on two real robots with dexterous hands and investigate the following questions:

- Can a policy learn tasks that have never been teleoperated from human video alone, and how does RoboTryOn compare with other retargeting or overlaying methods? (Table 1 and Figure 3)
- Does our synthesized data bridge the visual and action gaps between human videos and real robot data? (Figures 4 and 6 and Table 8)
- Can synthesized data substitute for teleoperated demonstrations, and how does success scale with the amount of human video? (Figure 5)

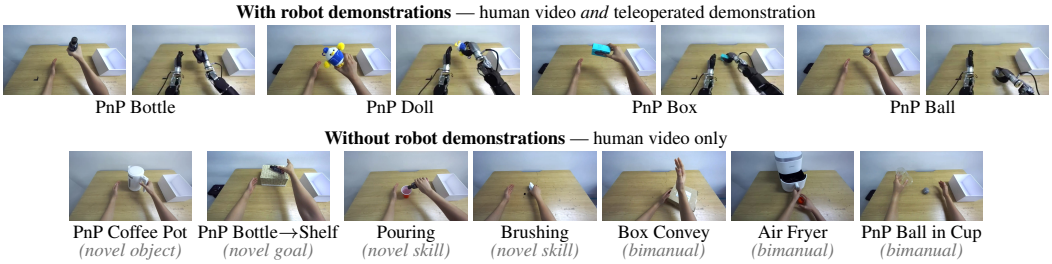


Figure 2: **Example of real-world benchmark tasks.** For the four PnP tasks (top) both a human video and a teleoperated demonstration exist; for the other seven (bottom) only the human video exists.

4.1 Experimental Setups

Robots and tasks. We evaluate RoboTryOn on two bimanual robots with dexterous hands: an OpenArm with two 7-DoF arms and two 6-DoF Inspire RH56F1 hands, and an RB-Y1 with two 7-DoF arms and two 20-DoF WujiHand2 hands. Both are evaluated on eleven tabletop tasks (see Figure 2). There are 4 pick-and-place (PnP) tasks, *i.e.*, in-domain tasks, with 50 teleoperated demonstrations per robot and 50 human videos each: *PnP Bottle*, *Doll*, *Box*, and *Ball*. The other 7 tasks, *i.e.*, out-of-domain tasks, have 50 human videos each and no robot demonstration; they introduce a single novel object task (*PnP Coffee Pot*), a single novel goal task (*PnP Bottle* $\rightarrow$ *Shelf*), two novel skill tasks (*Pouring*, *Brushing*), and three bimanual manipulation tasks (*Box Convey*, *Air Fryer*, *PnP Ball in Cup*), whereas the teleoperated tasks are all single-arm tasks. Further details on the robots and tasks are provided in Section C.1.

Baselines. We compare with the two existing ways of converting human video into robot training data. *EgoMimic* (Kareer et al., 2025) co-trains on the human frames with the arm masked out, using actions retargeted from the reconstructed hand pose; *Masquerade* (Lepert et al., 2025a), a representative overlay method (Lepert et al., 2024; 2025b; Wang et al., 2026; Liu et al., 2026), inpaints the arm out, composites a rendering of the robot in the retargeted pose, and adds an auxiliary loss on future 2D robot keypoints. Following their protocols, both apply the same edit to the robot data and to the live camera image at test time, whereas RoboTryOn is deployed on the raw camera view (see details in Section B.3). All methods share the policy model, the teleoperated episodes, and the human videos.

Implementation details. We fine-tune $\pi_{0.5}$ (Black et al., 2025) on the 200 teleoperated episodes, on the 550 synthesized episodes, or on all 750 together, for the same number of training steps in every case. Each policy is evaluated over 28 trials per task with randomized object positions (see details in Section C.1). The image model, the video model, and the IDM are trained on the same corpus of teleoperated robot episodes, the demonstrations of the four in-domain PnP tasks plus additional recordings on the OpenArm, the RB-Y1, and the GR-1. None of the additional recordings is used to train any policy. Their backbones and hyperparameters are given in Section B.2.8.

4.2 Main Results

Table 1 reports success on the OpenArm and the RB-Y1, and Figure 3 shows one human video translated to target robots.

Learning tasks without robot demonstrations. RoboTryOn lets the policy learn tasks for which no robot demonstration exists, including bimanual ones. On the seven tasks that have only human video, the policy trained on teleoperation alone reaches 3.1% on the OpenArm and 1.0% on the RB-Y1, succeeding only when a familiar object happens to be grasped. Adding the episodes synthesized by RoboTryOn raises success to 73.0% and 46.9%, and a policy trained on the synthe-

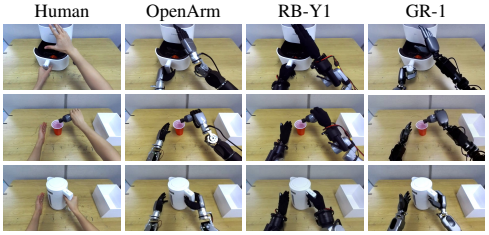


Figure 3: **One human video, three robots.** Human videos (left) translated into the OpenArm, the RB-Y1, and the GR-1 by a single set of models; only the embodiment changes.

Table 1: **Real-world results on (a) OpenArm and (b) RB-Y1.** Success rate (%) over 28 trials per task. In-domain: the four PnP tasks with robot demonstrations; out-of-domain: the seven tasks with human video only, grouped by what they add. "+" rows add the converted human videos to the teleoperated demonstrations; the last row uses the synthesized episodes alone. Baselines are evaluated following their own protocols. Per-task results for the out-of-domain tasks are provided in Section F.2.

(a) OpenArm Benchmark Results.

Method	Data Source		In-domain	Out-of-domain				
	Robot	Human		Novel Object	Novel Goal	Novel Skill	Bimanual	Avg.
Teleop only	✓	✗	47.3	14.3	7.1	0.0	0.0	3.1
+ EgoMimic (Kareer et al., 2025)	✓	✓	50.0	14.3	0.0	10.7	0.0	5.1
+ Masquerade (Lepert et al., 2025a)	✓	✓	67.9	21.4	7.1	14.3	0.0	8.2
+ RoboTryOn (Ours)	✓	✓	**75.9**	**64.3**	**89.3**	**83.9**	**63.1**	**73.0**
RoboTryOn (Ours)	✗	✓	57.1	50.0	57.1	58.9	38.1	48.5

(b) RB-Y1 Benchmark Results.

Method	Data Source		In-domain	Out-of-domain				
	Robot	Human		Novel Object	Novel Goal	Novel Skill	Bimanual	Avg.
Teleop only	✓	✗	69.6	7.1	0.0	0.0	0.0	1.0
+ EgoMimic (Kareer et al., 2025)	✓	✓	63.4	14.3	0.0	10.7	0.0	5.1
+ Masquerade (Lepert et al., 2025a)	✓	✓	63.4	21.4	7.1	12.5	0.0	7.7
+ RoboTryOn (Ours)	✓	✓	**79.5**	60.7	**67.9**	**57.1**	**28.6**	**46.9**
RoboTryOn (Ours)	✗	✓	60.7	**64.3**	60.7	53.6	16.7	40.3

sized episodes alone reaches 48.5% and 40.3%. The gain extends to the three bimanual tasks, which the policy learns from human video even though every teleoperated episode is single-arm, reaching 63.1% on the OpenArm and 28.6% on the RB-Y1. The two robots differ in kinematics, hand, and camera placement, yet a single image model, video model, and IDM serve both (see Figure 3).

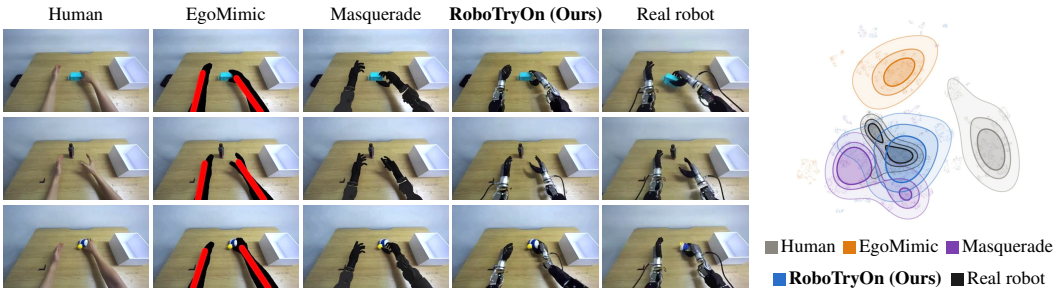
Augmenting tasks with robot demonstrations. The synthesized episodes also improve tasks that already have robot demonstrations. On the four PnP tasks, where 50 teleoperated demonstrations per task exist, they serve as additional demonstrations that cover more object placements and more intermediate states along the motion. Success rises from 47.3% to 75.9% on the OpenArm and from 69.6% to 79.5% on the RB-Y1. The synthesized episodes alone reach 57.1% and 60.7%, on par with the teleoperated demonstrations on the RB-Y1 and above them on the OpenArm.

Comparison with retargeting and overlay. Given the same human videos and robot data, RoboTryOn outperforms EgoMimic and Masquerade on both the in-domain and the out-of-domain tasks. On the seven tasks without robot demonstrations, EgoMimic reaches 5.1% on both robots and Masquerade 8.2% and 7.7%. On the bimanual tasks both score zero, never moving the second arm and attempting a single-arm pick-and-place on whatever object is placed, the one skill the teleoperated data contains. On the four PnP tasks they add little over teleoperation alone, with EgoMimic at 50.0% and 63.4% and Masquerade at 67.9% and 63.4%. Two gaps explain the difference. Their retargeted actions are less accurate than our action labels, and the frames they train on are edited in a way that obscures the dexterous fingers, so they remain apart from real robot frames (Section 4.3). Moreover, the same edit must also be applied to the live camera image at test time, whereas RoboTryOn is deployed on the raw camera view. Rollouts of the policies are shown in Section F.4.

4.3 More Analysis

We further analyze how close the synthesized data is to real robot data and what human video is worth as a source of demonstrations. Throughout this section we use the OpenArm with $\pi_{0.5}$.

Visual gap. RoboTryOn closes the visual gap, whereas masking and overlay do not (Figure 4). We measure how far each kind of frame is from a real robot frame, as the policy's own encoder sees it. As shown in Figure 4, the human frames and their two edited versions each form a cluster of their own, apart from the real robot frames, whereas the frames synthesized by RoboTryOn fall among the real ones. The Fr'echet SigLIP distance (Heusel et al., 2017; Zhai et al., 2023) to the real frames confirms this. Masking the arm leaves a frame as far from the robot as the raw video, overlaying a rendered robot brings it about halfway (58.2), and RoboTryOn brings it four times closer than



(a) Qualitative comparison (b) Feature distribution

Figure 4: **Visual gap to real robot video.** (a) The same moments as recorded, as EgoMimic and Masquerade edit them, and as RoboTryOn translates them, next to similar moments from teleoperation. (b) UMAP of SigLIP features of $\pi_{0.5}$ with density contours; equal frame counts per corpus, real robot frames from the four PnP tasks and the others from all eleven.

the raw video (32.7 against 125.5). This is why both baselines must apply the same edit to the live camera image at test time, whereas RoboTryOn is deployed on the raw view.

Action gap. Our action labels succeed about twice as often as retargeted ones when replayed (Figure 6c), and the difference lies at the grasp. The baselines label a human video by retargeting the reconstructed hand onto the robot (Qin et al., 2023) and solving the arm by inverse kinematics, whereas RoboTryOn obtains action labels from the synthesized robot video with the IDM. Human video carries no ground truth for the robot action, so we judge a set of action labels by replaying it open loop in a simulated replica of the scene (Figure 6a; Section D). Details of the simulation setup and tasks are provided in Section C.2. As shown in Figure 6c, our action labels complete the task in 69.7% of the replays against 34.8% for the retargeted ones. The hand pose reconstructed from the video is imprecise, and mapping it onto a robot hand of a different shape adds further error, so the robot hand arrives near the object but fails to close a firm grasp. RoboTryOn instead uses the reconstructed hand only as guidance. The image and video models render the robot hand grasping the object as the robot data shows, with the object where it lies in the video, so the generated frames contain a grasp that holds, and the IDM turns that grasp into action labels (Figure 6b). The reconstruction enters the action labels only through the wrist position, which we refine with the stereo views before the correction (see details in Section B.2.2).

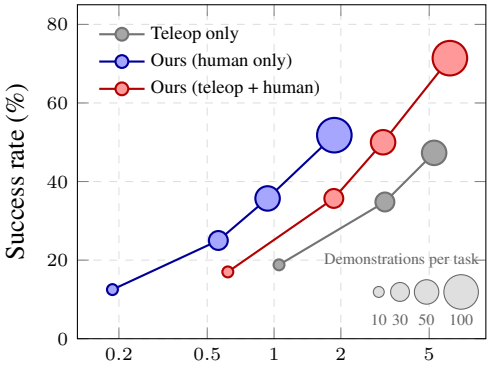


Figure 5: **Collection cost.** Success rate versus person-hours on four tasks. Marker size indicates demonstrations per task.

Scaling and cost. Human video alone reaches the success of the 50 teleoperated demonstrations at about a third of their cost, and the two sources add up (Figure 5). Collecting a teleoperated episode takes about five times as long as recording a human video (see Table 10). Figure 5 plots success on the PnP tasks against this cost. Per demonstration, a translated human video is worth a little less than a teleoperated episode, but per hour of collection it is worth far more. Adding as many human videos as there are teleoperated demonstrations raises success by 24 points for less than one additional hour of recording.

Execution speed. Retiming the hand skeleton flow gives direct control over the execution speed of the resulting policy, and matching the pace of the teleoperation data also raises success (Table 3). The human videos of our tasks were recorded about $1.25\times$ faster than the teleoperated episodes (see details in Section B.2.4). By default we synthesize from the video as recorded, and Table 3 compares this with retiming the skeleton flow to the pace of the teleoperation data, or to twice that pace, and synthesizing the video again from the same keyframes. The completion time of the policy follows the pace of its data. Retiming to the teleoperation pace raises success on the PnP tasks, where the synthesized episodes then match the teleoperated ones in pace. Retiming to twice that pace lowers

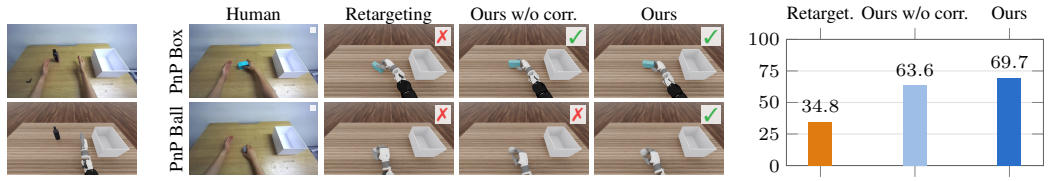
Table 2: **Ablation results on OpenArm.** Success rates (%) are averaged over four in-domain and seven out-of-domain tasks.

(a) Component ablation.

Method	In-dom.	Out-of-dom.
RoboTryOn (Ours)	**75.9**	**73.0**
w/o Autoregressive reference	68.8	66.3
w/o Hand skeleton flow	42.0	21.4
w/o Frame interpolation	42.9	39.3
w/o IDM wrist correction	65.2	48.0

(b) Policy backbone comparison.

Method	In-dom.	Out-of-dom.
$\pi_{0.5}$ (Teleop only)	47.3	3.1
+ **RoboTryOn (Ours)**	**75.9**	**73.0**
RLDX-1-Img (Teleop only)	53.6	0.0
+ **RoboTryOn (Ours)**	**60.7**	**40.3**



(a) Real to sim (b) Grasp moment (c) Replay success (%)

Figure 6: **Action gap.** (a) A real scene and its simulated replica. (b) The grasp moment of one human video under the three action labelings, replayed open loop. (c) Replay success over 66 episodes of three PnP tasks (per task in Table 8).

success on both task groups but yields the fastest policy. The pace of the data thus trades success for execution speed.

4.4 Ablations

Every component contributes, and the skeleton flow matters most. Table 2a removes one component of each stage at a time, the autoregressive reference in keyframe translation, the hand skeleton flow in video synthesis, the intermediate and last keyframes, and the wrist correction of the action labels (see details in Section F.3). Without the skeleton flow, the video model interpolates between keyframes without guidance, and the motion no longer follows the demonstrator. Generating from the first keyframe alone keeps the motion, but the objects and the scene drift away from the human video, so the episode no longer performs the task. Without the autoregressive reference, the robot stays consistent, but the scene becomes discontinuous between keyframes. Removing the wrist correction lowers success far more on the out-of-domain tasks than on the in-domain tasks, where the IDM predicts novel motions and the second arm less precisely. Table 2b repeats the main comparison with RLDX-1-Img, a VLA of a different architecture, and finds the same gain from the synthesized episodes, so the gain is a property of the data rather than of a particular policy. Per-task results are given in Tables 12 and 13.

Table 3: **Execution speed.** Success rate (%) on OpenArm and mean time from rollout start to object placement on the four PnP tasks.

Method	Speed	In-dom.	Out-of-dom.	Time (s)
Teleop only	1×	47.3	3.1	14.2
Ours (as recorded)	1.25×	75.9	**73.0**	11.9
Ours (retimed)	1×	**85.7**	70.4	13.8
	2×	57.1	42.9	**7.4**

5 Conclusion

We have presented RoboTryOn, which turns an egocentric human video into a robot episode the way virtual try-on turns a photograph of a person into one in a different garment. The demonstrated motion is kept and the embodiment is swapped. Translated keyframes, a video model guided by the hand skeleton, and a stereo inverse dynamics model with a wrist correction from the hand reconstruction yield robot video with action labels, from models trained on robot data alone. On two real robots with dexterous hands, a policy learns seven tasks for which no teleoperated demonstration exists, above EgoMimic and Masquerade and without their test-time edit of the camera view. On tasks with demonstrations, the synthesized episodes act as augmentation, and human video reaches the result of teleoperation at about a third of its collection cost. Limitations and future work are discussed in Section A.

Reproducibility Statement

The three models, their conditions, and their objectives are specified in Sections 3.1 to 3.4, and the backbones in Section 4.1. Section B gives the keyframe selection, the skeleton and mask construction, the training and inference configurations with their compute, the training corpus, the policy training recipe, the baseline protocols, the real-world environment, and the evaluation protocol. The visual gap measurement and the simulation replay of Section 4.3 are described in Section D. All backbones are publicly available models, and the full training and inference configurations are listed in Tables 4 and 5.

Ethics Statement

The human videos used in this work were recorded by the authors. They show hands and a tabletop, and faces that appear in a frame are blurred. All robot data were collected on our own robots. The policies were executed on the real robots under human supervision with joint and velocity limits in place. Since the method can turn any video of a person into robot data, we intend it for videos recorded by consenting demonstrators. The authors declare no conflicts of interest.

AI Use Statement

We used a large language model to assist with editing the text and the LaTeX source, including figure and table layout, and with routine scripting. All AI-assisted text was reviewed by the authors against the experiments and data, AI-assisted code was reviewed and tested by the authors, and all reported results were produced and verified by the authors. The authors take full responsibility for the content of this paper.

References

Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video PreTraining (VPT): Learning to act by watching unlabeled online videos. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, brian ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/black25a.html.

Black Forest Labs. FLUX.2 [klein]: Towards Interactive Visual Intelligence. https://bfl.ai/blog/flux2-klein-towards-interactive-visual-intelligence, 2026.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Xindong He, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

Claire Chen, Zhongchun Yu, Hojung Choi, Mark Cutkosky, and Jeannette Bohg. DexForce: Extracting force-informed actions from kinesthetic demonstrations for dexterous manipulation. *arXiv preprint arXiv:2501.10356*, 2025.

Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-TeleVision: Teleoperation with immersive active visual feedback. In *Conference on Robot Learning (CoRL)*, 2024.

Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. In *Robotics: Science and Systems (RSS)*, 2024.

Seunghwan Choi, Sunghyun Park, Minsoo Lee, and Jaegul Choo. Viton-hd: High-resolution virtual try-on via misalignment-aware normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Yisol Choi, Sangkyung Kwak, Kyungmin Lee, Hyungwon Choi, and Jinwoo Shin. Improving diffusion models for authentic virtual try-on in the wild. In *European Conference on Computer Vision (ECCV)*, 2024.

Hai Ci, Xiaokang Liu, Pei Yang, Yiren Song, and Mike Zheng Shou. H2R-Grounder: A paired-data-free paradigm for translating human interaction videos into physically grounded robot videos. *arXiv preprint arXiv:2512.09406*, 2025.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. Rescaling egocentric vision: Collection, pipeline and challenges for EPIC-KITCHENS-100. *International Journal of Computer Vision (IJCV)*, 2022.

Yilun Du, Mengjiao Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Joshua B Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Dyna Robotics. Dyna-2: A 1-million-hour scaling law for world-action models. https://www.dyna.co/dyna-2, 2026. Technical report.

Zixun Fang, Wei Zhai, Aimin Su, Hongliang Song, Kai Zhu, Mao Wang, Yu Chen, Zhiheng Liu, Yang Cao, and Zheng-Jun Zha. Vivid: Video virtual try-on using diffusion models. *arXiv preprint arXiv:2405.11794*, 2024.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, Qianli Ma, Seungjun Nah, Loic Magne, Jiannan Xiang, Yuqi Xie, Ruijie Zheng, Dantong Niu, You Liang Tan, K. R. Zentner, George Kurian, Suneel Indupuru, Pooya Jannaty, Jinwei Gu, Jun Zhang, Jitendra Malik, Pieter Abbeel, Ming-Yu Liu, Yuke Zhu, Joel Jang, and Linxi Fan. DreamDojo: A generalist robot world model from large-scale human videos. In *International Conference on Machine Learning (ICML)*, 2026.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The "something something" video database for learning and evaluating visual common sense. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-Exo4D: Understanding skilled human activity from first- and third-person perspectives. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, Eitan Richardson, Guy Shiran, Itay Chachy, Jonathan Chetboun, Michael Finkelson, Michael Kupchick, Nir Zabari, Nitzan Guetta, Noa Kotler, Ofir Bibi, Ori Gordon, Poriya Panet, Roi Benita, Shahar Armon, Victor Kulikov, Yaron Inger, Yonatan Shiftan, Zeev Melumian, and Zeev Farbman. LTX-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.

Ankur Handa, Karl Van Wyk, Wei Yang, Jacky Liang, Yu-Wei Chao, Qian Wan, Stan Birchfield, Nathan Ratliff, and Dieter Fox. DexPilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Ryan Hoque, Peide Huang, David J. Yoon, Mouli Sivapurapu, and Jian Zhang. EgoDex: Learning dexterous manipulation from large-scale egocentric video. *arXiv preprint arXiv:2505.11709*, 2025.

Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, Loic Magne, Ajay Mandlekar, Avnish Narayan, You Liang Tan, Guanzhi Wang, Jing Wang, Qi Wang, Yinzhen Xu, Xiaohui Zeng, Kaiyuan Zheng, Ruijie Zheng, Ming-Yu Liu, Luke Zettlemoyer, Dieter Fox, Jan Kautz, Scott Reed, Yuke Zhu, and Linxi Fan. Dreamgen: Unlocking generalization in robot learning through video world models. In *Conference on Robot Learning (CoRL)*, 2025.

Boyuan Jiang, Xiaobin Hu, Donghao Luo, Qingdong He, Chengming Xu, Jinlong Peng, Jiangning Zhang, Chengjie Wang, Yunsheng Wu, and Yanwei Fu. Fitdit: Advancing the authentic garment details for high-fidelity virtual try-on. *arXiv preprint arXiv:2411.10499*, 2024.

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 *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13226–13233. IEEE, 2025.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. DROID: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems (RSS)*, 2024.

Dongyoung Kim, Huiwon Jang, Myungkyu Koo, Suhyeok Jang, Taeyoung Kim, et al. Rldx-1 technical report. *arXiv preprint arXiv:2605.03269*, 2026.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning (CoRL)*, 2024.

Chuizheng Kong, Yunho Cho, Wonsuhk Jung, Idris Wibowo, Parth Shinde, Sundhar Vinodh-Sangeetha, Long Kiu Chung, Zhenyang Chen, Andrew Mattei, Advaith Nidumukkala, Alexander Elias, Danfei Xu, Taylor Higgins, and Shreyas Kousik. A closed-form geometric retargeting solver for upper body humanoid robot teleoperation, 2026. URL https://arxiv.org/abs/2602.01632.

Seungyong Lee and Jeong-gi Kwak. Voost: A unified and scalable diffusion transformer for bidirectional virtual try-on and try-off. In *SIGGRAPH Asia 2025 Conference Papers*, 2025.

Marion Lepert, Ria Doshi, and Jeannette Bohg. Shadow: Leveraging segmentation masks for cross-embodiment policy transfer. In *Conference on Robot Learning (CoRL)*, 2024.

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

Marion Lepert, Jiaying Fang, and Jeannette Bohg. Phantom: Training robots without robots using only human videos. In *Conference on Robot Learning (CoRL)*, 2025b.

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 manipulation with real-life human activity videos. *arXiv preprint arXiv:2510.21571*, 2025.

Zhen Li, Cheng-Ze Lu, Jianhua Qin, Chun-Le Guo, and Ming-Ming Cheng. Towards an end-to-end framework for flow-guided video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Lightricks. LTX-2.5. https://huggingface.co/Lightricks/LTX-2.5, 2026.

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 (ICLR)*, 2023.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. RDT-1B: A diffusion foundation model for bimanual manipulation. In *International Conference on Learning Representations (ICLR)*, 2025a.

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 (ICLR)*, 2023.

Yangcen Liu, Woo Chul Shin, Yunhai Han, Zhenyang Chen, Harish Ravichandar, and Danfei Xu. Immimic: Cross-domain imitation from human videos via mapping and interpolation. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 834–858. PMLR, 27–30 Sep 2025b. URL https://proceedings.mlr.press/v305/liu25b.html.

Yangcen Liu, Shuo Cheng, Xinchen Yin, Woo Chul Shin, Alfred Cueva, Yiran Yang, Zhenyang Chen, Chuye Zhang, and Danfei Xu. EgoEngine: From egocentric human videos to high-fidelity dexterous robot demonstrations. *arXiv preprint arXiv:2606.12604*, 2026.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21013–21022, June 2022.

Hao Luo, Yicheng Feng, Wanpeng Zhang, Sipeng Zheng, Ye Wang, Haoqi Yuan, Jiazheng Liu, Chaoyi Xu, Qin Jin, and Zongqing Lu. Being-h0: Vision-language-action pretraining from large-scale human videos. *arXiv preprint arXiv:2507.15597*, 2025.

Arjun Majumdar, Karmesh Yadav, Sergio Arnaud, Yecheng Jason Ma, Claire Chen, Sneha Silwal, Aryan Jain, Vincent-Pierre Berges, Pieter Abbeel, Jitendra Malik, Dhruv Batra, Yixin Lin, Oleksandr Maksymets, Aravind Rajeswaran, and Franziska Meier. Where are we in the search for an artificial visual cortex for embodied intelligence? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Mayank Mittal, Calvin Yu, Qinxi Yu, Jingzhou Liu, Nikita Rudin, David Hoeller, Jia Lin Yuan, Ritvik Singh, Yunrong Guo, Hammad Mazhar, Ajay Mandlekar, Buck Babich, Gavriel State, Marco Hutter, and Animesh Garg. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6):3740–3747, 2023.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2022.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Pannag Sanketi, Quan Vuong, Ted Xiao, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems (RSS)*, 2024.

Open X-Embodiment Collaboration. Open X-Embodiment: Robotic learning datasets and RT-X models. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

Ryan Punamiya, Dhruv Patel, Patcharapong Aphiwetsa, Pranav Kuppili, Lawrence Zhu, Simar Kareer, Judy Hoffman, and Danfei Xu. Egobridge: Domain adaptation for generalizable imitation from egocentric human data. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 47092–47122. Curran Associates, Inc., 2025. doi: 10.52202/085713-1569. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/434756a76c44e220d5f32d6e0b488c0c-Paper-Conference.pdf.

Ryan Punamiya, Simar Kareer, Zeyi Liu, Josh Citron, Ri-Zhao Qiu, Xiongyi Cai, Alexey Gavryushin, Jiaqi Chen, Davide Liconti, Lawrence Y. Zhu, Patcharapong Aphiwetsa, Baoyu Li, Aniketh Cheluva, Pranav Kuppili, Yangcen Liu, Dhruv Patel, Aidan Gao, Hye-Young Chung, Ryan Co, Renee Zbizika, Jeff Liu, Xiaomeng Xu, Haoyu Xiong, Geng Chen, Sebastiano Oliani, Wenkai Xuan, Chenyu Yang, Xi Wang, James Fort, Richard Newcombe, Josh Gao, Jason Chong, Garrett Matsuda, Aseem Doriwala, Marc Pollefeys, Robert Katzschmann, Xiaolong Wang, Shuran Song, Judy Hoffman, and Danfei Xu. Egoverse: An egocentric human dataset for robot learning from around the world, 2026. URL https://arxiv.org/abs/2604.07607.

Yuzhe Qin, Wei Yang, Binghao Huang, Karl Van Wyk, Hao Su, Xiaolong Wang, Yu-Wei Chao, and Dieter Fox. AnyTeleop: A general vision-based dexterous robot arm-hand teleoperation system. In *Robotics: Science and Systems (RSS)*, 2023.

Abraham Savitzky and Marcel J. E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8):1627–1639, 1964.

Dandan Shan, Jiaqi Geng, Michelle Shu, and David F. Fouhey. Understanding human hands in contact at internet scale. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Yiren Song, Cheng Liu, Weijia Mao, and Mike Zheng Shou. Mitty: Diffusion-based human-to-robot video generation. *arXiv preprint arXiv:2512.17253*, 2025.

Chen Wang, Haochen Shi, Weizhuo Wang, Ruohan Zhang, Li Fei-Fei, and C. Karen Liu. DexCap: Scalable and portable mocap data collection system for dexterous manipulation. In *Robotics: Science and Systems (RSS)*, 2024.

Ye Wang, Pei Lin, Xiong-Hui Chen, Haoqi Yuan, Zhixuan Liang, Yiyang Huang, Anzhe Chen, Zixing Lei, Jie Zhang, Tao Zhang, Haoyang Li, Tong Zhang, Chenxi Xiao, Ziyuan Jiao, and Qin Jin. Ego2Robot: Scalable robot data synthesis from egocentric human data. *arXiv preprint arXiv:2608.02580*, 2026.

Philipp Wu, Yide Shentu, Zhongke Yi, Xingyu Lin, and Pieter Abbeel. GELLO: A general, low-cost, and intuitive teleoperation framework for robot manipulators. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024.

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. *arXiv preprint arXiv:2505.21864*, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything V2. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

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. *arXiv preprint arXiv:2507.12440*, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

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

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

Shangchen Zhou, Chongyi Li, Kelvin C.K. Chan, and Chen Change Loy. Propainter: Improving propagation and transformer for video inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Tongchun Zuo, Zaiyu Huang, Shuliang Ning, Ente Lin, Chao Liang, Zerong Zheng, Jianwen Jiang, Yuan Zhang, Mingyuan Gao, and Xin Dong. Dreamvvt: Mastering realistic video virtual try-on in the wild via a stage-wise diffusion transformer framework. *arXiv preprint arXiv:2508.02807*, 2025.

A Limitations and Future Work

Limitations. Adding an embodiment still requires a few hours of teleoperated video of that robot, on any tasks, to teach the generative models and the inverse dynamics model what it looks like and how it moves; the cost is paid once per robot rather than once per task, but it is not zero. The actions are pseudo-labels, joints read from pixels and a wrist position read from the hand reconstruction, so tasks that depend on forces, or on precision below what stereo video and a hand reconstruction resolve, such as tight insertion, are outside what RoboTryOn can currently label, and the wrist correction relies on the head-camera calibration. The human video is recorded from where the robot's camera will be; in-the-wild egocentric video translates qualitatively (see Figure 3), but a policy trained on it would face a moving viewpoint that ours does not. The demonstrated motion must be one the robot can execute: the pipeline can retime the human video to the robot's pace, but it does not correct a reach outside the robot's workspace or a grasp the robot hand cannot form. Our evaluation covers tabletop manipulation on two robots, and the GR-1 is used only as a translation target. Finally, synthesis trades collection time for compute (see Table 10), which favours tasks that are expensive to teleoperate.

Future work. The natural next step is scale: translating large egocentric corpora such as EgoDex into every embodiment we own, so that a robot's training data grows with the supply of human video rather than with the supply of operators. Closing the loop, where the policy's own rollouts are fed back to improve the inverse dynamics model; learning the fusion of pixel-based and hand-based action estimates rather than fixing it at the wrist; and extending the interface beyond hands to tools and deformable objects are the directions we find most promising.

B Implementation Detail

B.1 Policy Training Detail

Both $\pi_{0.5}$ and RLDX-1-Img are fully fine-tuned for 30k steps with a batch size of 64 on four H100 GPUs, with the same settings for every data mixture of Table 1. $\pi_{0.5}$ is trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, weight decay 0.01) at a learning rate of 2.5×10^{-5} with cosine decay to 2.5×10^{-6}, gradient clipping at norm 1.0, and an action chunk of 50 steps (2.5 s); states and actions are normalized by their 1st and 99th percentiles to $[-1, 1]$. RLDX-1-Img follows the configuration of its official repository: the vision-language backbone is frozen except for its top four language layers, the action expert is trained in full (2.4B of 6.9B parameters in total), the learning rate is 10^{-4} with cosine decay, and the action chunk is 16 steps (0.8 s); the optimizer and the normalization are the same as for $\pi_{0.5}$.

B.2 RoboTryOn Pipeline Details

B.2.1 Keyframe Selection

Keyframes are selected from the 3D wrist trajectories of the reconstructed hands (Section B.2.2). Frames at which the wrist speed has a local minimum are candidates: the hands are unblurred there, and a hand has typically arrived at a new location. Among the candidates, the next keyframe is the first whose wrist positions differ from those of the previous keyframe by more than a threshold set from the distribution of wrist displacements over the episode, so that keyframes mark changes of location rather than fixed intervals. Because the reconstruction is most reliable when a hand is slow and fully visible, this rule also tends to avoid frames in which a hand is undetected or detected with low confidence. The base video model supports clips of up to 20 seconds, but for the consistency of the synthesized video we cap the interval between consecutive keyframes at 145 frames, about 7 seconds at 20 Hz. When no candidate falls within the cap, the frames with the highest reconstruction confidence are considered first and the one with the largest displacement among them is taken, and failing that the frame at the cap or the last frame of the video. Pick-and-place videos yield about four keyframes and the longer tasks six to seven.

B.2.2 HAND SKELETON

Hand reconstruction. The human hands are reconstructed with HaWoR (Zhang et al., 2025) from each view of the stereo camera. The depth of the wrist is corrected by triangulating the two views, and the left and right hand predictions of the two views complement each other, so that a hand that leaves the frame in one view and is misdetected there is recovered from the other view.

Keypoint set. Each hand is represented by 16 keypoints: the knuckle, the middle joint, and the tip of each of the five fingers, and the wrist. The skeleton connects the three keypoints of each finger in sequence and connects the knuckles to one another and to the wrist. The same set is defined for the human hand, the Inspire RH56F1, the WujiHand2, and the GR-1 hand, so that hands with different degrees of freedom map to the same keypoints. For a robot the keypoints come from forward kinematics of the recorded joint state, projected into each camera view; for a person they come from the reconstructed hand, retargeted to the target robot hand. Because the retargeting is imprecise, keypoints are perturbed and randomly dropped during training: each keypoint is displaced by up to half the length of the finger segment it belongs to, the whole skeleton is dropped with probability 0.2, and each keypoint is dropped independently with probability 0.2. The model thus learns to treat the skeleton as a coarse guide while its prior over robot hands, learned from the robot data, determines the configuration the generated hand takes. The retargeting error does not enter the action labels, which are read from the generated video by the IDM. The skeleton is rendered as its 2D projection, drawn over the frame for the image model and as a video of the same length for the video model.

B.2.3 KEYFRAME TRANSLATION

Mask construction. The mask is shape-agnostic in the sense of virtual try-on: it is enlarged and smoothed so that the model cannot recover the original silhouette of the arm and hand it covers. For a robot frame the mask is built from a rendering of the robot from its URDF at the recorded joint state and a SAM3 segmentation of the robot, which are complementary, since neither alone covers the robot exactly. During training the mask is further deformed to random sizes and shapes so that the model is robust to its size and outline, and, with some probability and where the object is known, the region of the manipulated object is removed from the mask. At inference the mask of the human arm and hand comes from SAM3 alone, dilated so that the shape cannot be inferred, and the region of the manipulated object is always removed so that the object is preserved.

Reference sampling. The reference frame is sampled from the same episode two to five seconds before the target; a closer frame would allow the model to copy the reference instead of learning to move the robot.

Stereo generation. Both views are generated as one token sequence with an additional rotary position dimension separating left and right tokens; training mixes single-view and stereo examples.

Autoregressive inference. The denoised latent of $\hat{\mathbf{o}}^r_{t-}$ is reused directly as the reference tokens for $\hat{\mathbf{o}}^r_t$, without decoding and re-encoding.

First keyframe. The first keyframe has no previously translated keyframe to serve as reference, so its reference is a real frame of the target robot with everything but the robot masked out, which selects the embodiment. To train for this case, the reference is replaced by such a frame with probability 0.05 during training.

B.2.4 VIDEO SYNTHESIS

All videos and actions are recorded, synthesized, and labeled at 20 Hz. A demonstration is cut at keyframe times into clips of at most 145 frames (Section B.2.1); each clip is synthesized with the one to four keyframes that fall inside it, of which the first frame of the clip is always one, its first frame is the last generated frame of the previous clip, and the clips are concatenated in time. The rendered skeleton flow is encoded by the video VAE like the clip itself and attended to as in-context tokens. Demonstrations of 10–25 seconds are synthesized in this way (Figure 13).

Table 4: **Training configuration.** Resolutions are per view.

	Image model f_θ	Video model g_ϕ	IDM h_ψ
Base model	FLUX.2 [klein] 9B Base	LTX-2.5	Depth Anything V2-L
Trained parameters	full	LoRA (rank 64)	vision tower + action head
Input resolution	1024×576	512×288, then 1024×576	224×224
Views	stereo : mono = 3 : 1	stereo + mono	stereo (GR-1: mono)
Clip length / chunk	–	≤145 frames @ 20 fps	20 frames @ 20 fps
Steps	10k	4k + 1k	60k
Global batch size	64	16	128
Learning rate	10^{-5}, cosine	10^{-4}	10^{-4}, cosine
Optimizer	AdamW (0.9, 0.999)	AdamW (0.9, 0.999)	AdamW (0.9, 0.999)
Hardware	4×H100	4×H100, then 4×H200	4×H200

B.2.5 Retiming.

By default the human video is synthesized as recorded. For Table 3 each human video is divided by its action into phases, approaching the object, grasping it, lifting it, and placing it, into which the tasks without robot demonstrations divide as well. For each phase the skeleton flow is resampled in time so that its speed matches the mean action speed of the teleoperated episodes in that phase, at the chosen multiple, and the video is synthesized again from the same keyframes at their new temporal positions. Human demonstrators grasp and lift faster than the teleoperators in many episodes, so these are the phases the retiming changes most.

B.2.6 Inverse Dynamics Model

The IDM takes frames t and $t{+}20$ (1 s at 20 fps) at 224×224, stereo for the OpenArm and RB-Y1 and monocular for the GR-1, and predicts a 20-step chunk of the full joint vector (28, 66, and 44 dimensions). Following the DreamGen IDM (Jang et al., 2025), the three embodiments share one model and are distinguished by an embodiment tag, and the action head is a flow-matching head on the Depth Anything V2 backbone, whose vision tower is fine-tuned with its last layer and head frozen. For labeling, the model is applied in a sliding window, overlapping predictions are averaged, and the sequence is smoothed with a Savitzky–Golay filter (Savitzky & Golay, 1964) of polynomial order 2 over a window of 17 frames. The states $\hat{\mathbf{q}}_t$ of a synthesized demonstration are derived from its action labels by the relation measured on the teleoperated data, where the recorded state lags the commanded action by four steps, so that $\hat{\mathbf{q}}_t = \hat{\mathbf{a}}_{t-4}$.

B.2.7 End-effector correction.

The human wrist position is taken from the hand reconstruction of Section B.2.2, in the camera frame, and transformed into the robot base frame with the head-camera calibration, since the human video is recorded from the position of the robot's head camera. The robot hand implied by the IDM's joints is scaled, finger segment by finger segment, to the size of the reconstructed human hand, and the target wrist position is the one at which the end effector of the scaled hand coincides with the reconstructed wrist. Since the reconstruction is noisy, the target is taken only from segments with high reconstruction confidence and smoothed in time, so that the correction moves the arm gradually rather than snapping it to the reconstruction. The target orientation is the wrist orientation obtained by forward kinematics from the IDM's arm joints, so that only the translation is corrected. The arm joints are then solved by inverse kinematics in MuJoCo, initialized from the IDM's arm joints of the same frame; the hand joints are left unchanged. The correction is applied to both arms for bimanual demonstrations. Section 4.4 ablates it.

B.2.8 Training Setup

Training Corpus All three models, the image model, the video model, and the IDM, are trained on the same corpus of teleoperated robot episodes: 3.75 hours (756 episodes) collected on the OpenArm with the Inspire RH56F1 hand, 1.47 hours (221 episodes) on the RB-Y1 with the WujiHand2, and 2.50 hours (1,180 clips) of the GR-1 taken from the public GR00T teleoperation dataset released with DreamDojo (Gao et al., 2026). The OpenArm and RB-Y1 recordings are stereo; the GR-1 recordings are monocular, so the generative models learn stereo generation from the first two. The

Table 5: **Inference configuration and cost per unit.** Time is measured on a single H100. Video stage 2 upsamples and refines the output of stage 1.

Stage	Model	Output per view	Views	Steps	Time (s)
Hand reconstruction	HaWoR	3D hand pose, per frame	2	–	0.448 / frame
Keyframe translation	f_θ	1024×576	2	30	34.6 / keyframe
Video synthesis, stage 1	g_ϕ (distilled)	512×288	2	8	78.2 / clip
Video synthesis, stage 2	g_ϕ	1024×576	2	3	78.2 / clip
Action labeling	h_ψ	20-step joint chunk	2	–	0.077 / window
End-effector correction	MuJoCo IK	arm joints, per frame	–	–	0.22 / frame

Table 6: **Pipeline wall-clock time** for a 15 s demonstration (300 frames, about four keyframes, one embodiment) on a single H100.

Stage	Units	Time (s)
Hand reconstruction + keyframe selection	450 frames	202
Keyframe translation	5 keyframes	173
Video synthesis, stage 1	3 clips	150
Video synthesis, stage 2	3 clips	150
Action labeling	281 windows	22
End-effector correction	300 frames	66

corpus consists of the teleoperated demonstrations of the four pick-and-place tasks used in Section 4 plus additional teleoperated recordings; the additional recordings are used only to train the three models above and play no part in training any VLA policy.

Training Configuration The image model f_θ is a fully fine-tuned FLUX.2 [klein] 9B Base (Black Forest Labs, 2026), the video model g_ϕ an in-context LoRA on LTX-2.5 (Lightricks, 2026; HaCohen et al., 2026), and the IDM h_ψ a Depth Anything V2 backbone (Yang et al., 2024) with the action head of the DreamGen IDM (Jang et al., 2025). Table 4 lists their training configuration, and all three are trained on the corpus of Section B.2.8. The image model is trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and a cosine learning rate schedule. Its input embedder does not accept the channel-wise concatenation of the conditioned frame with the noisy target, so the embedder is extended by zero-initialized input channels that are trained with the rest of the model. Training examples are stereo and single-view in a 3:1 ratio; the GR-1 recordings are single-view only. The video model LoRA has rank 64 and is trained with AdamW ($\beta_1 = 0.9$, $\beta_2 = 0.999$) in two stages matching the two-stage inference of Section B.4: 4k steps at 512×288 per view followed by 1k steps at 1024×576 per view.

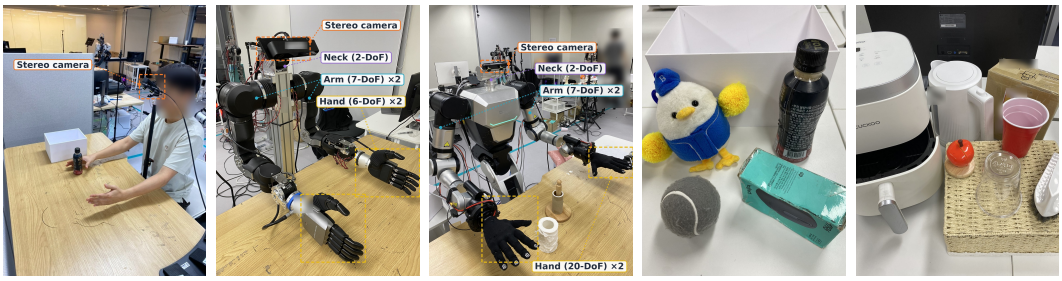
B.3 Baselines

Baselines Both baselines are re-implemented on our hardware with the same backbone, teleoperated episodes, and human videos as RoboTryOn.

Retargeting. Both use the same pseudo-actions: the 3D hand pose reconstructed with HaWoR is mapped to the robot hand by a geometric joint-angle mapping (finger flexion angles matched to the hand's joint ranges, wrist pose transformed into the base frame with the head-camera calibration) following AnyTeleop (Qin et al., 2023), and the arm joints are obtained by inverse kinematics in MuJoCo from the wrist pose.

EgoMimic. The human arm and hand are masked in every human frame; in the robot episodes the robot arm is masked from the recorded state, by projecting the URDF into the camera. At test time the mask is placed on the live image from the current robot state and the camera calibration. The masking follows the official implementation.

Masquerade. The human arm is inpainted out and a rendering of the robot from its URDF in the retargeted state is composited onto the frame; robot episodes receive the same rendering from the recorded state, and at test time the rendering is composited onto the live image from the current state. The original uses E2FGVI (Li et al., 2022) for the inpainting; we use ProPainter (Zhou et al., 2023), which leaves fewer artifacts on our videos. Following the original, the policy is trained with



(a) Human video collection (b) OpenArm + RH56F1 (c) RB-Y1 + WujiHand2 (d) In-domain objects (e) Out-of-domain objects

Figure 7: **Real-world setup.** (a) Human video collection from the robot's camera position. (b) OpenArm embodiment. (c) RB-Y1 embodiment. (d) Objects of the in-domain tasks. (e) Objects introduced for the out-of-domain tasks.

an auxiliary loss that predicts future 2D robot hand keypoints from the image features, with targets from the retargeted (human) or recorded (robot) state; the head, the prediction horizon, and the loss weight follow the official implementation. The original single-arm gripper pipeline is extended to both arms and to the dexterous hands of our robots.

B.4 Inference Details

Inference Configuration and Compute Table 5 lists the inference configuration of every stage and its cost on a single H100. The video model runs in two stages: the first generates the clip at 512×288 per view with a distilled model in 8 sampling steps, and the second upsamples the result to 1024×576 per view and refines it in 3 steps. Both stages are conditioned on the same keyframes and skeleton flow, the second at the higher resolution, and both generate the two stereo views in one sequence. The image model generates each keyframe at 1024×576 per view in 30 sampling steps with a guidance scale of 3.5, and the keyframes of a demonstration are generated sequentially since each conditions on the previous one. The IDM predicts a 20-step action chunk from each 1 s window and is applied in a sliding window with a stride of one frame over the demonstration.

Table 6 breaks down the wall-clock time of the full pipeline for a typical 15 s demonstration (300 frames at 20 fps, about four keyframes, translated into one embodiment). Table 10 accounts collection in person-hours and Section A notes the trade of collection time for compute.

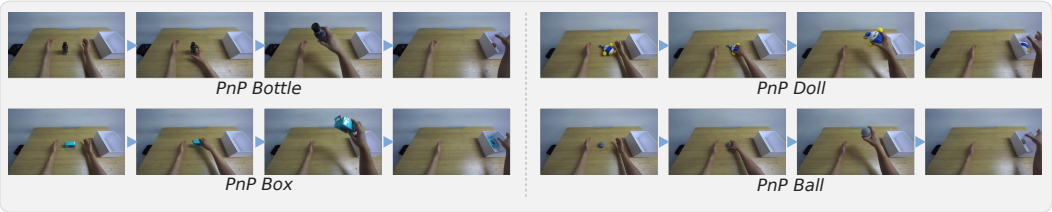
C Experimental Details

C.1 Real-World Environment

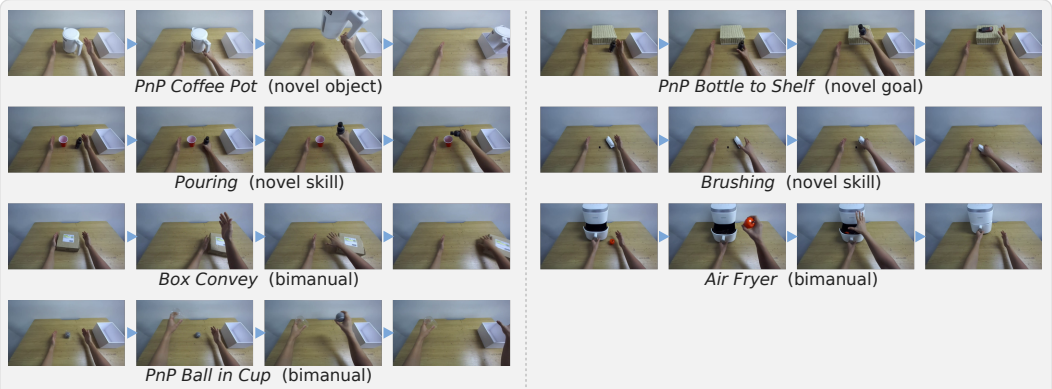
Hardware Setup. We use two humanoid robots, OpenArm and RB-Y1 (Figure 7b,c), in our real-world experiments. Both humanoids are equipped with two 7-DoF arms. Each arm of OpenArm is equipped with a 6-DoF Inspire RH56F1 hand, and each arm of RB-Y1 is equipped with a 20-DoF WujiHand2 hand. The head of each robot is mounted on a 2-DoF neck that supports pitch and yaw motions. A stereo camera is mounted on each neck, and its left and right egocentric views are used as visual inputs throughout all experiments.

Task Details. All tasks are formulated as language-conditioned manipulation tasks. We evaluate eleven tasks in total (Figure 2), comprising four in-domain pick-and-place (PnP) tasks and seven out-of-domain tasks. The in-domain tasks (Figure 2a) consist of *PnP Bottle*, *PnP Doll*, *PnP Box*, and *PnP Ball*, with 50 teleoperated demonstrations per robot embodiment and 50 human videos for each task. The human videos are recorded from the position of the robot's head camera (Figure 7a). The seven out-of-domain tasks (Figure 2b) have 50 human videos each and no robot demonstrations: *PnP Coffee Pot*, *PnP Bottle to Shelf*, *Pouring*, *Brushing*, *Box Convey*, *Air Fryer*, and *PnP Ball in Cup*.

The four in-domain PnP tasks share the same task structure, where the robot grasps a designated object and places it into a white container (Figure 7d). *PnP Coffee Pot* follows the same pick-and-



(a) In-domain tasks



(b) Out-of-domain tasks

Figure 8: **Real-world tasks.** (a) In-domain tasks with teleoperated demonstrations and human videos. (b) Out-of-domain tasks with human videos only.

place structure with a novel coffee pot object. *PnP Bottle to Shelf* requires the robot to grasp a bottle and place it at a designated location on a shelf. *Pouring* requires the robot to grasp a bottle and pour its contents into a red cup. *Brushing* requires the robot to grasp a brush and wipe the area marked with black tape. *Box Convey* requires the robot to move a box to the right using both hands sequentially, with the left hand moving it first and the right hand moving it further. *Air Fryer* requires the robot to open an air fryer, grasp an apple-shaped object, place it inside, and close the door. Finally, *PnP Ball in Cup* requires the robot to lift an inverted cup with the left hand, grasp the ball with the right hand, and place it into the target container. Figure 7e shows the objects introduced by the out-of-domain tasks.

Evaluation Protocol We conduct 28 evaluation trials for each task. For single-arm tasks, a trial receives a score of 1 only when the task is successfully completed. For bimanual tasks, each hand contributes 0.5 to the score, yielding a maximum score of 1 when both required hand actions are successfully completed. The initial positions of the target objects are randomized within the workspace for each trial. For the PnP tasks, including *PnP Bottle*, *PnP Doll*, *PnP Box*, *PnP Ball*, *PnP Coffee Pot*, and *PnP Bottle to Shelf*, a trial is considered successful when the object is placed at the designated goal location. For *Pouring*, the trial is successful when the bottle is tilted over the target cup. For *Brushing*, success is determined by whether the brush reaches the designated region marked with black tape. For *Box Convey*, the trial is successful when the specified sequential manipulation with the left and right hands is correctly executed. For *Air Fryer*, success requires opening the air fryer, placing the apple-shaped object inside, and closing the door. For *PnP Ball in Cup*, success requires lifting the cup with the left hand, grasping the ball with the right hand, and placing the ball at the designated goal location.

C.2 Simulation Environment

Setup. We build a simulated counterpart of the OpenArm setup in Isaac Lab Mittal et al. (2023). The simulated robot follows the real OpenArm: two 7-DoF arms with 6-DoF Inspire RH56F1 hands and a 2-DoF neck, fixed at the downward pitch measured on the real robot. The head camera is positioned to match the mounting configuration of the real stereo camera, ensuring that the simulated egocentric view shares the same viewpoint as the real one (Figure 9). Actions are fingertip and wrist

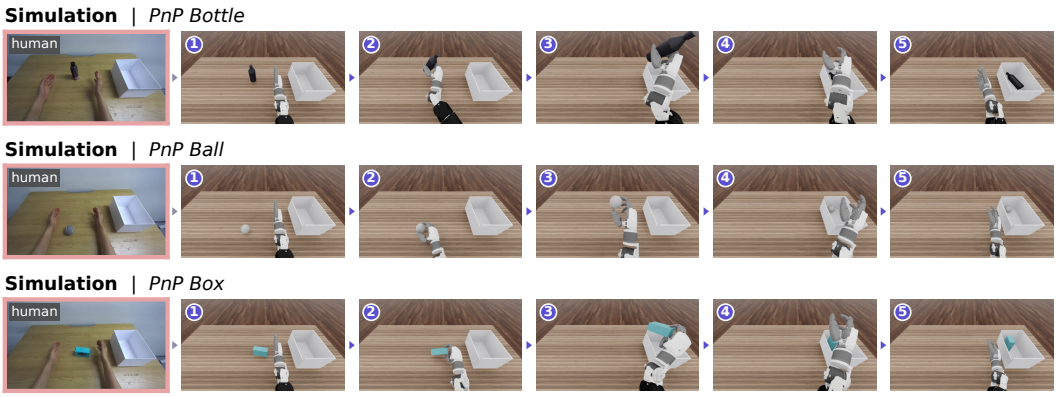


Figure 9: **Simulated tasks.** Each row shows the human video of a task (left) and a successful episode in simulation (1–5): the right hand grasps the object, lifts it, and places it in the container.

Table 7: **Fréchet SigLIP distance to the real robot frames.** Lower is closer. Real robot frames are from the four PnP tasks; each corpus is compared on all eleven tasks (as in Figure 4), on the four PnP tasks alone, and on the seven tasks without robot data alone.

Corpus	All eleven tasks	Four PnP tasks	Seven tasks w/o robot data
Human	125.5	120.1	133.5
EgoMimic	129.0	127.1	134.6
Masquerade	58.2	50.3	68.8
RoboTryOn (Ours)	**32.7**	**26.6**	**41.9**

keypoint targets for the right hand at 20 Hz, converted to arm joint commands by inverse kinematics and to finger joint commands.

Tasks. We simulate the three in-domain pick-and-place tasks that have simulated object counterparts, *PnP Bottle*, *PnP Ball*, and *PnP Box*; *PnP Doll* is excluded because no simulation asset of the doll exists. As in the real setup, the robot grasps the object and places it into the white container (Figure 9). Initial object positions are drawn from four pick locations arranged in a 2×2 grid on the desk.

Evaluation protocol. Each task is evaluated on 22 episodes (66 in total), spread over the four pick locations. An episode runs for 60 s without retries and counts as a success if the object is inside the container at the end; an object that falls in and bounces out counts as a failure.

D Analysis Detail

Visual Gap Analysis From each corpus, 5,000 frames are sampled uniformly: the human videos, their EgoMimic and Masquerade edits, the RoboTryOn translations, and the teleoperated robot episodes. Each frame is encoded by the pretrained SigLIP vision tower (Zhai et al., 2023) of $\pi_{0.5}$ and its features are mean-pooled over image tokens. Figure 4(b) is a UMAP projection fitted on the union of the five sets. Table 7 gives the Fréchet distance (Heusel et al., 2017) between the SigLIP features of each corpus and those of the real robot frames, once on the frames of Figure 4, once restricted to the four PnP tasks that have robot data, and once to the seven tasks that do not. The frames of the seven tasks without robot data lie closer to the real frames than the edited frames of the four tasks with it, so the overlap in Figure 4 is not a recollection of the training episodes.

Action Replay in Simulation The three PnP scenes (Bottle, Box, Ball; Doll is excluded because no simulation asset of it exists) are rebuilt in MuJoCo with the OpenArm model and the object meshes placed at the positions recorded in the real scene, measured with the depth sensor and the stereo views of the ZED 2i head camera and verified by overlaying the first frame of the video on the simulated scene. For each human video the labels are executed open loop from the same initial configuration, and an episode counts as a success when the object is picked up and placed at the

Table 8: **Action gap, per task.** Success (%) of replaying the action labels of the same human videos open loop, without a policy, in a simulated replica of the real scene with the objects at their recorded positions (OpenArm; 22 episodes per task); averages in Figure 6(c).

Labeling	PnP Bottle	PnP Box	PnP Ball	**Avg.**
Retargeting (Qin et al., 2023)	63.6	31.8	9.1	34.8
RoboTryOn (Ours) w/o wrist correction	63.6	**72.7**	54.5	63.6
RoboTryOn (Ours)	**77.3**	63.6	**68.2**	**69.7**

Table 9: **Success against collection cost, per condition.** Data behind Figure 5. Each policy is trained on the four PnP tasks alone with N demonstrations per task and evaluated over 28 trials per task (112 in total). Person-hours follow Table 10: 38 teleoperated episodes or 214 human videos per person-hour; the equal split uses $N/2$ of each.

Training data	N / task	Person-hours	Successes / 112	SR (%)
Teleop only	10	1.05	21	18.8
	30	3.16	39	34.8
	50	5.26	53	47.3
Human video only	10	0.19	14	12.5
	30	0.56	28	25.0
	50	0.93	40	35.7
	100	1.87	58	51.8
Teleop + human (1:1)	10	0.62	19	17.0
	30	1.86	40	35.7
	50	3.10	56	50.0
	100	6.20	80	**71.4**

target of the task. *Retargeting*: the reconstructed hand (Zhang et al., 2025) is mapped to the robot hand (Qin et al., 2023) and the arm joints are solved by inverse kinematics from the wrist pose, the labels EgoMimic and Masquerade train on. *RoboTryOn*: the IDM labels of the synthesized video, with and without the wrist correction of Section 3.4. Each labeling is replayed on the same 22 episodes of each task; on Bottle, Box, and Ball, RoboTryOn succeeds on 17, 14, and 15 of them, RoboTryOn without the wrist correction on 14, 16, and 12, and retargeting on 14, 7, and 2 (Table 8).

E Flow-Matching Objective and Sampling

The policy and the three models of RoboTryOn are conditional flow-matching models (Lipman et al., 2023; Liu et al., 2023). For the policy, with $\mathbf{A}_t^\tau = \tau \mathbf{A}_t + (1-\tau)\,\epsilon$, $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and $\tau \sim \mathcal{U}[0,1]$, the objective is

$$\mathcal{L}_{\text{policy}}(\omega) = \mathbb{E}_{(\mathbf{o}_t, \mathbf{q}_t, \mathbf{A}_t, \mathbf{c}) \sim \mathcal{D},\, \epsilon,\, \tau} \left[\left\| \pi_\omega\big(\mathbf{A}_t^\tau, \tau;\ \mathbf{o}_t, \mathbf{q}_t, \mathbf{c}\big) - (\mathbf{A}_t - \epsilon) \right\|_2^2 \right], \tag{6}$$

and Eqs. (1), (3), and (5) have the same form with their own targets and conditions; the image and video models predict in the latent space of a pretrained encoder $\mathcal{E}$. A trained model predicts a velocity field, and a sample $\mathbf{x} \sim f_\theta(\cdot \mid \mathbf{y})$ under condition $\mathbf{y}$ is obtained by integrating this field from Gaussian noise at $\tau = 0$ to $\tau = 1$; the same holds for π_ω, g_ϕ, and h_ψ.

F Additional Results

F.1 Data Collection Cost

Table 10 lists the cost model behind the person-hour axis of Figure 5, and Table 9 the eleven policies plotted there. Teleoperation occupies two people and yields about 14 minutes of demonstration per person-hour; a single person records about 45 minutes of human video in the same time and completes each task faster, so the gap in episode count (38 against 214 per person-hour) is larger than the gap in footage. The teleop-only policy with 50 demonstrations per task is the *Real* policy of Table 1a; the other ten policies are trained for this analysis on the four PnP tasks alone, so their values are not those of Table 1a, whose policies also train on the seven tasks without robot demonstrations.

Table 10: **Data collection cost.** Teleoperation occupies two people, so one person-hour yields 14 min of robot demonstration against 45 min of egocentric human video; because a person also performs each task faster than a teleoperated robot, the gap in episode count is 5–6×.

	Robot Teleoperation	Human Ego Video
People per session	2	1
Demonstration footage / person-hour	14 min	45 min
Episodes / person-hour	38	214
Hardware	Robot + teleoperation system	Fixed camera
Trained operator required	Yes	No
Occupies robot	Yes	No
Bimanual overhead	High (two arms to operate)	None

Table 11: **Per-task results behind Table 1.** Success rates (%) over 28 trials per task using $\pi_{0.5}$ on OpenArm and RB-Y1. Robot demonstrations are available only for the PnP tasks in (a). "+" rows add the human videos converted by each method to the teleoperated demonstrations; **RoboTryOn (Ours)** only uses the synthesized episodes alone. EgoMimic and Masquerade are deployed with their test-time mask and overlay; RoboTryOn uses the raw camera view.

(a) Tasks with robot demonstrations

	Data		In-domain tasks				
Method	Teleop	Human	PnP Bottle	PnP Doll	PnP Box	PnP Ball	Avg.
OpenArm							
Teleop only	✓	✗	71.4	39.3	42.9	35.7	47.3
+ EgoMimic (Kareer et al., 2025)	✓	✓	64.3	50.0	42.9	42.9	50.0
+ Masquerade (Lepert et al., 2025a)	✓	✓	**92.9**	42.9	**71.4**	64.3	67.9
+ RoboTryOn (Ours)	✓	✓	**92.9**	**71.4**	**71.4**	**67.9**	**75.9**
RoboTryOn (Ours)	✗	✓	82.1	57.1	50.0	39.3	57.1
RB-Y1							
Teleop only	✓	✗	82.1	64.3	75.0	57.1	69.6
+ EgoMimic	✓	✓	67.9	71.4	64.3	50.0	63.4
+ Masquerade	✓	✓	67.9	57.1	67.9	60.7	63.4
+ RoboTryOn (Ours)	✓	✓	**89.3**	**75.0**	**85.7**	**67.9**	**79.5**
RoboTryOn (Ours)	✗	✓	71.4	64.3	75.0	32.1	60.7

(b) Tasks without task-specific robot demonstrations

	Data		Novel object	Novel goal	Novel skill		Bimanual			
Method	Teleop	Human	PnP Coffee Pot	PnP Bottle to Shelf	Pouring	Brushing	Box Convey	Air Fryer	PnP Ball in Cup	Avg.
OpenArm										
Teleop only	✓	✗	14.3	7.1	0.0	0.0	0.0	0.0	0.0	3.1
+ EgoMimic	✓	✓	14.3	0.0	7.1	14.3	0.0	0.0	0.0	5.1
+ Masquerade	✓	✓	21.4	7.1	7.1	21.4	0.0	0.0	0.0	8.2
+ RoboTryOn (Ours)	✓	✓	**64.3**	**89.3**	**85.7**	**82.1**	**78.6**	**60.7**	**50.0**	**73.0**
RoboTryOn (Ours) only	✗	✓	50.0	57.1	64.3	53.6	57.1	35.7	21.4	48.5
RB-Y1										
Teleop only	✓	✗	7.1	0.0	0.0	0.0	0.0	0.0	0.0	1.0
+ EgoMimic	✓	✓	14.3	0.0	10.7	10.7	0.0	0.0	0.0	5.1
+ Masquerade	✓	✓	21.4	7.1	10.7	14.3	0.0	0.0	0.0	7.7
+ RoboTryOn (Ours)	✓	✓	60.7	**67.9**	**75.0**	39.3	**32.1**	0.0	**53.6**	**46.9**
RoboTryOn (Ours) only	✗	✓	**64.3**	60.7	60.7	**46.4**	28.6	0.0	21.4	40.3

F.2 Per-task Results

Table 11 gives the per-task results for the main experiments. Table 12 gives the per-task results for the two policy backbones, $\pi_{0.5}$ and RLDX-1-Img (Kim et al., 2026), behind Table 2b. Table 13 gives the per-task results behind Table 2a.

Table 12: **Policy backbone, per task.** Success rate (%) on the OpenArm for each backbone, trained on the teleoperated episodes alone and with the episodes synthesized by RoboTryOn.

		In-domain					Out-of-domain							
Backbone	Data	PnP Bottle	PnP Doll	PnP Box	PnP Ball	Avg.	PnP Coffee Pot	PnP Bottle to Shelf	Pouring	Brushing	Box Convey	Air Fryer	PnP Ball in Cup	Avg.
$\pi_{0.5}$	Teleop only	71.4	39.3	42.9	35.7	47.3	14.3	7.1	0.0	0.0	0.0	0.0	0.0	3.1
	+ RoboTryOn (Ours)	**92.9**	**71.4**	**71.4**	**67.9**	**75.9**	**64.3**	**89.3**	**85.7**	**82.1**	**78.6**	**60.7**	**50.0**	**73.0**
RLDX-1-Img	Teleop only	50.0	**100.0**	57.1	7.1	53.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	+ RoboTryOn (Ours)	**57.1**	57.1	**64.3**	**64.3**	**60.7**	**28.6**	**21.4**	**42.9**	**32.1**	**64.3**	**21.4**	**71.4**	**40.3**

Table 13: **Component ablation, per task.** Success rate (%) on the OpenArm with $\pi_{0.5}$ for the variants of Table 2a.

	In-domain					Out-of-domain							
Variant	PnP Bottle	PnP Doll	PnP Box	PnP Ball	Avg.	PnP Coffee Pot	PnP Bottle to Shelf	Pouring	Brushing	Box Convey	Air Fryer	PnP Ball in Cup	Avg.
RoboTryOn (Ours) (full)	92.9	71.4	71.4	67.9	75.9	64.3	89.3	85.7	82.1	78.6	60.7	50.0	73.0
w/o autoregressive reference	85.7	64.3	64.3	60.7	68.8	57.1	82.1	78.6	75.0	71.4	53.6	46.4	66.3
w/o hand skeleton flow	50.0	39.3	39.3	39.3	42.0	10.7	42.9	42.9	28.6	14.3	7.1	3.6	21.4
w/o intermediate and last frame	53.6	39.3	39.3	39.3	42.9	21.4	17.9	53.6	60.7	53.6	39.3	28.6	39.3
w/o IDM wrist corr.	50.0	64.3	**78.6**	**67.9**	65.2	35.7	53.6	46.4	75.0	71.4	32.1	21.4	48.0

F.3 ABLATED VARIANTS.

Figure 10 shows failure cases of the ablated variants on *PnP Coffee Pot*. Without the autoregressive reference, each keyframe is translated independently, resulting in inconsistent grasp configurations: the fingers wrap the handle from below at 5.0 s and reach behind it at 6.0 s, causing visible hand jitter between keyframes. In contrast, the full model maintains a consistent grasp throughout.

Without interpolation, the video is generated from the first keyframe alone. Although the hand still follows the skeleton, the coffee pot changes into a box inside the container at 11.0 s. Without skeleton flow, the motion between keyframes deviates from the demonstrator's trajectory, causing the wrist and hand to lose the configuration required to grasp the handle at 3.5 s.

F.4 QUALITATIVE RESULTS

F.4.1 ROLLOUTS ON OPENARM

In Figure 11, we show all rollouts on the OpenArm from the policy trained on both teleoperated and synthesized episodes. Figure 12(a) shows the four tasks with robot demonstrations, while Figure 12(b) shows the seven tasks with human video only.

F.4.2 ROLLOUTS ON RB-Y1

In Figure 12, we show all rollouts on the RB-Y1 from the policy trained on both teleoperated and synthesized episodes. Figure 12(a) shows the four tasks with robot demonstrations, while Figure 12(b) shows the seven tasks with human video only.

No method completes *Air Fryer* on the RB-Y1 (Table 11). The task requires inserting the fingers between the handle and the fryer body before closing, a fine-grained grasping motion not observed in the teleoperated data and beyond the resolution of the IDM labels. The longer, multi-joint fingers of the WujiHand2 also tend to push the fryer away when closing in front of the handle, unlike the shorter one-DoF fingers of the Inspire hand.

F.4.3 LONG HORIZON SYNTHESIS

Figure 13 shows a 25-second human video and the corresponding long-horizon robot video generated from its selected keyframes. The generated video is divided into clips at the keyframe times, with the last generated frame of each clip used as the first frame of the next. This preserves the consistency of the robot and objects across clip boundaries.

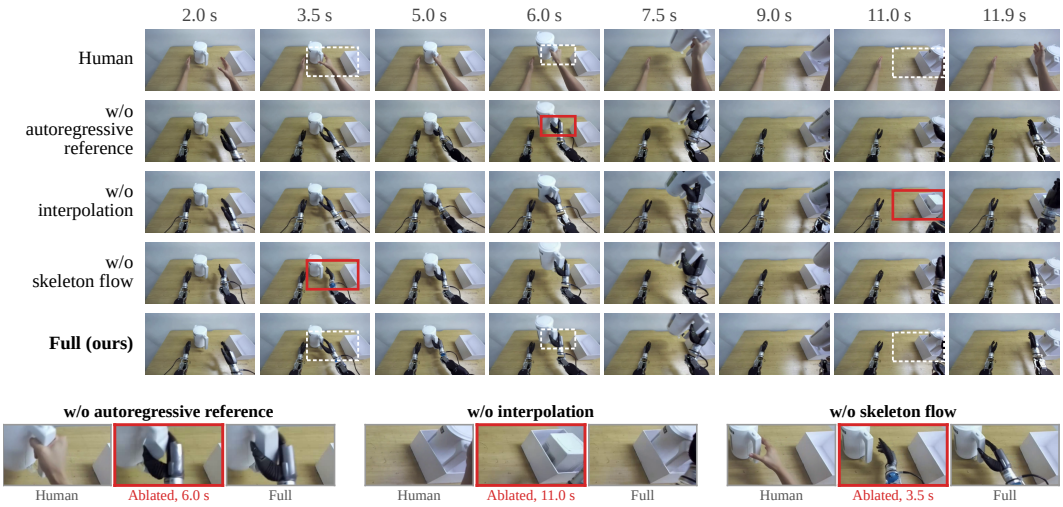
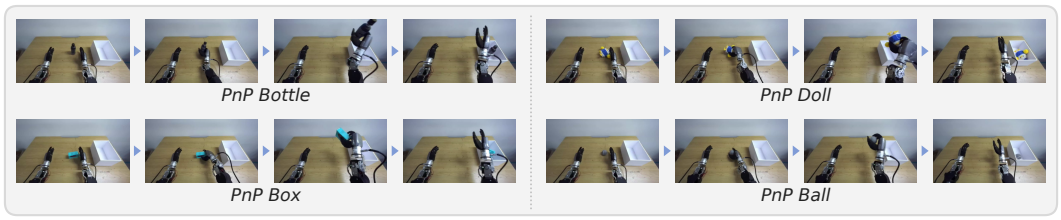


Figure 10: **Failure cases of the ablated variants.** Human video and videos synthesized by the variants in Table 2a at the same time steps. Boxed regions are enlarged at representative failure points.

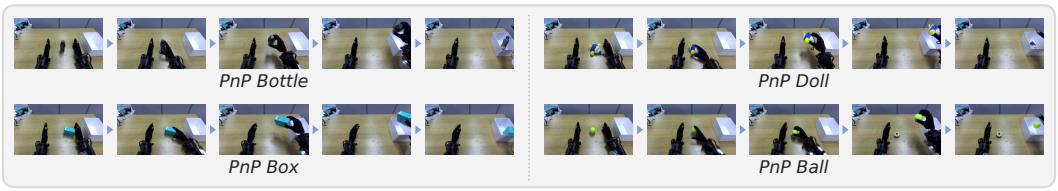


(a) In-domain tasks



(b) Out-of-domain tasks

Figure 11: **Rollouts on all eleven tasks on the OpenArm** With the policy trained on the teleoperated and synthesized episodes: (a) the four tasks with robot demonstrations and (b) the seven with human video only.



(a) In-domain tasks



(b) Out-of-domain tasks

Figure 12: **Rollouts on the RB-Y1**, with the policy trained on the teleoperated and synthesized episodes: (a) the four tasks with robot demonstrations and (b) the tasks with human video only. *Air Fryer* is omitted because no policy completes it on the RB-Y1 (0% success; Table 11);

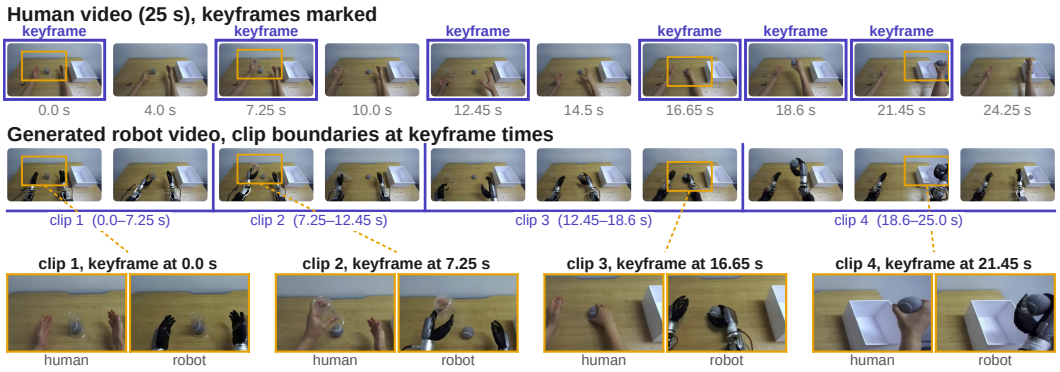


Figure 13: **Long-horizon synthesis.** A 25-second human video (top) with its selected keyframes, and the generated robot video (bottom) with the clip boundaries at the keyframe times. The last generated frame of a clip is the first frame of the next, so robot and objects stay consistent across boundaries.

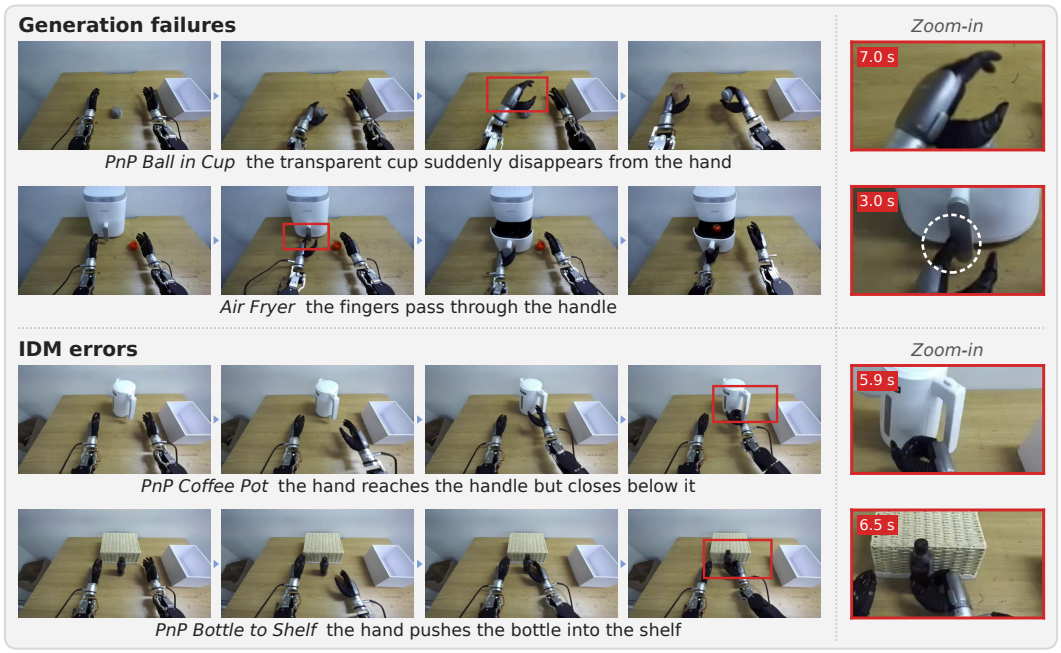


Figure 14: **Failure cases.** The red box in each row marks the region enlarged on the right. *Generation failures* (top): synthesized videos in which the generative models do not preserve the objects. On *PnP Ball in Cup* the transparent cup suddenly disappears from the hand as it is lifted, leaving the ball on the table; on *Air Fryer* the fingers pass through the handle instead of closing around it (dashed circle). *IDM errors* (bottom): policy rollouts that fail because the action labels of the synthesized episodes place the hand imprecisely. On *PnP Coffee Pot* the hand reaches the handle but closes below it, and on *PnP Bottle to Shelf* it closes low on the bottle and pushes it against the shelf instead of lifting it onto it.