RealDexUMI: A Wearable Universal Manipulation Interface for Dexterous Robot Learning

Author Names Omitted for Anonymous Review.

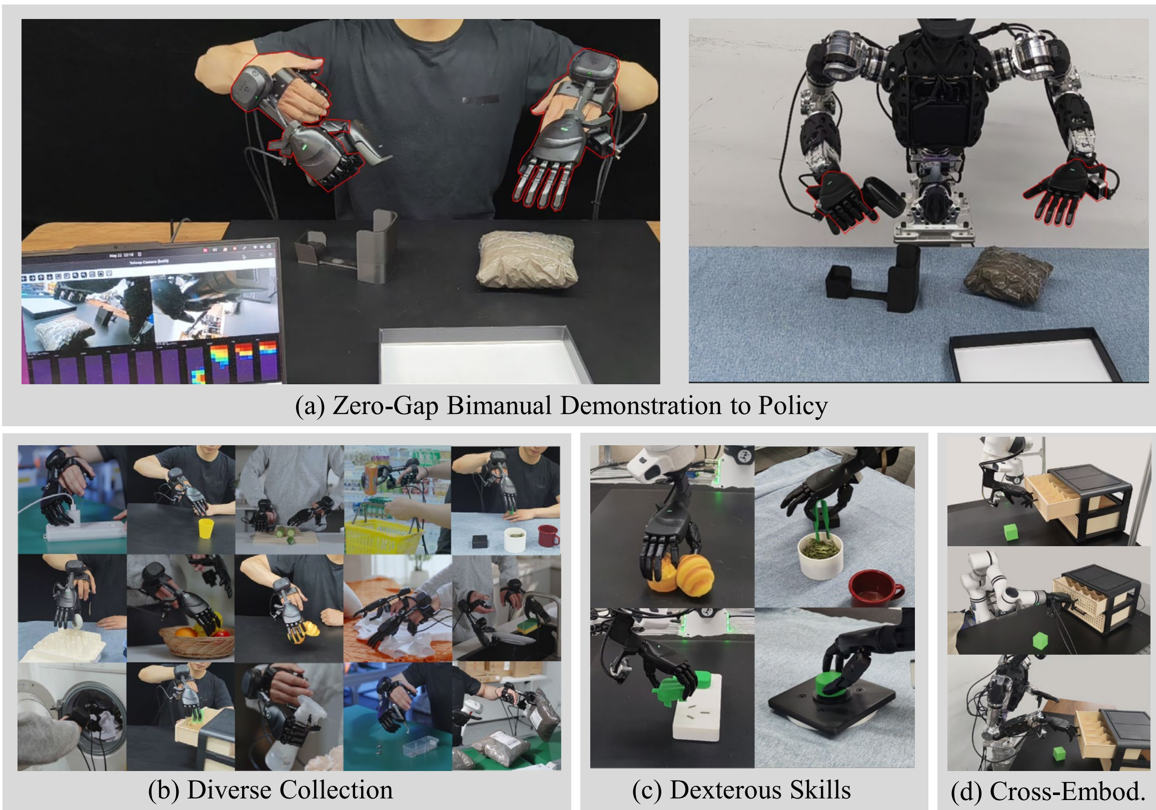


Fig. 1: RealDexUMI turns a dexterous hand into a shared interface for wearable demonstration and robot deployment: (a) system overview, (b) diverse task collection, (c) dexterous skill learning, and (d) cross-embodiment deployment.

***Abstract*—Learning dexterous manipulation requires demonstrations that preserve fine hand-object interactions while remaining executable at deployment. Existing pipelines either lose deployable dexterity through embodiment conversion or rely on robot-specific teleoperation that is costly to scale, and often lacks intuitive control for dexterous data collection. We present RealDexUMI, a wearable universal manipulation interface built around a shared dexterous end-effector module that integrates a lightweight dexterous hand, in-hand vision, and fingertip tactile sensing. A palm-side isomorphic teleoperation glove maps human finger inputs to robot-hand joint commands, enabling real-time, retargeting-free, intuitive, and precise hand control. The shared hand and sensing modules yield zero-gap end-effector data, with matched in-hand observations, tactile signals, contacts, and hand actions between collection and deployment. Across eight real-robot tasks spanning fine-grained, contact-rich, long-horizon, and bimanual manipulation, policies trained on RealDexUMI data achieve an average success rate of 88.75%, generalize to unseen initial poses, and transfer across three embodiments.**

Anonymous video: https://realdexumi.github.io/

I. Introduction

Human dexterity in demonstrations does not automatically translate into deployable robot dexterity. Many interfaces can capture dexterous human behavior, yet the resulting data may differ from the hand actions the robot can execute, the contacts it makes, the tactile signals it receives, and the observations available at deployment [3, 36, 34, 39, 10]. For dexterous manipulation, the relevant quantity is therefore not captured dexterity alone, but **deployable dexterity**: the extent to which demonstrated hand actions remain executable by the deployed dexterous end effector while the associated contacts, tactile signals, and observations are preserved. This distinction is especially important for contact-rich manipulation, where small mismatches in action execution, contact geometry, or local sensing can determine whether a skill succeeds or fails.

Existing data-collection pipelines preserve deployable dexterity only partially. UMI-style interfaces enable robot-free

TABLE I: Comparison of representative demonstration interfaces.

System	Act. DoF	Weight	Teleop complex.	Device size	Free wrist	Vision align	No retarg.	Tac.	A–S corr.
UMI [6]	1	780 g	Low	Small	✓	✓	✓	✗	✗
DexViTac [3]	6	<600 g	Low	Small	✓	✗	✗	✓	✗
DEXOP [9]	12/9/7	>1000 g	High	Large	✓	✓	✓	✓	✗
Dexexo [45]	6	>1000 g	High	Large	✗	✓	✓	✗	✗
DEX-Mouse [17]	6	>1000 g	High	Large	✗	✓	✓	✓	✓
Exo-viha [1]	6	1100 g	High	Large	✗	✓	✗	✗	✓
DexUMI [36]	12/6	850 g	Med.	Med.	✓	✗	✓	✓	✗
RealDexUMI	**6**	**680 g**	**Low**	**Med.**	✓	✓	✓	✓	✓

Note. Act. DoF denotes actuated end-effector DoF; slash-separated values indicate hardware variants. Teleop complex. is a survey-based estimate of operation difficulty, not hardware performance. Vision align denotes collection–deployment visual alignment without nontrivial post-processing. No retarg., Tac., and A–S corr. denote retargeting-free hand control, tactile sensing, and action–state correspondence.

data collection through end-effector-centric observations and actions [6, 28], with extensions to tactile sensing, additional viewpoints, mobile embodiments, and whole-body systems [19, 21, 22, 13, 46, 38, 4, 24, 41, 32, 20, 23, 37, 12, 40, 14]. However, most such systems use simple grippers with limited dexterity. Dexterous hand-motion interfaces capture richer human-hand behavior, but often require retargeting across different kinematics, contact surfaces, and sensing channels [3, 34, 30, 44, 7, 31, 2, 11, 5, 15, 26]. Exoskeleton and arm-attached systems improve robot-hand alignment, but can introduce bulky structures, restrict natural wrist motion, or record states rather than executable commands [9, 45, 33, 42, 18, 27, 16, 8, 29, 17, 1]. As summarized in Table I, prior interfaces trade off wearable operation, visual alignment, retargeting-free control, tactile sensing, and action–state correspondence.

RealDexUMI follows a different principle: preserve deployable dexterity by making the dexterous end effector a shared interface for both wearable demonstration and robot deployment. As illustrated in Fig. 1, the same interface supports robot-free dexterous task collection, policy learning from deployment-aligned demonstrations, and cross-embodiment deployment. During collection, object contact is made through the deployed robot hand itself, so the recorded actions, contacts, tactile signals, and in-hand observations remain aligned with deployment. This also preserves action–state correspondence: the measured hand state reflects what contact allows, while the paired executable command preserves the operator's intended correction. A hand-frame relative action representation further lets policies deploy across robot embodiments by changing only the robot-side IK and low-level controller.

We evaluate RealDexUMI across eight real-robot tasks spanning fine-grained, contact-rich, long-horizon, and bimanual manipulation. Policies trained from RealDexUMI demonstrations achieve an average full-task success rate of 88.75%, generalize to unseen initial poses, and transfer across three robot embodiments. Ablations further show that tactile observations and executable hand-command supervision are important for contact-rich dexterous policy learning.

We make three contributions:

- **A wearable interface for deployable dexterity.** We introduce RealDexUMI, a wearable manipulation interface built around a shared dexterous end-effector module and a palm-side isomorphic glove for real-time, retargeting-free control in the deployed hand's command space.
- **Deployment-aligned dexterous data.** The shared module preserves in-hand observations, tactile signals, contact surfaces, executable hand commands, and action–state correspondence from collection to deployment.
- **Cross-embodiment dexterous policy deployment.** Policies trained on RealDexUMI data use hand-frame relative actions to deploy the same checkpoints across heterogeneous robot embodiments.

II. RealDexUMI

RealDexUMI implements the shared end-effector interface with a wearable hardware system. As shown in Fig. 2, the system consists of a reusable dexterous end-effector module, a 6-DoF tracker, and a palm-side isomorphic teleoperation glove. The end-effector module integrates a lightweight dexterous hand, fingertip tactile sensors, and an in-hand camera. During collection, the operator wears and commands this module through the glove; during deployment, the same module is mounted on a robot arm.

A. Wearable End-Effector Interface

Unlike hand-motion interfaces that capture human-hand motion for retargeting, RealDexUMI collects demonstrations through the deployed robot hand itself. Object contacts, local visual observations, and tactile signals are therefore generated by the same surfaces and sensing layout used during robot execution. Each sensed glove DoF corresponds to one actuated hand DoF, producing a 6-D command vector in the deployed hand's executable command space. This direct linear mapping enables real-time, retargeting-free hand control while leaving the operator's fingers largely unobstructed.

B. Demonstration Data

Each demonstration records time-aligned in-hand RGB observations, fingertip tactile signals, measured hand states, executable hand commands, and 6-DoF end-effector poses.

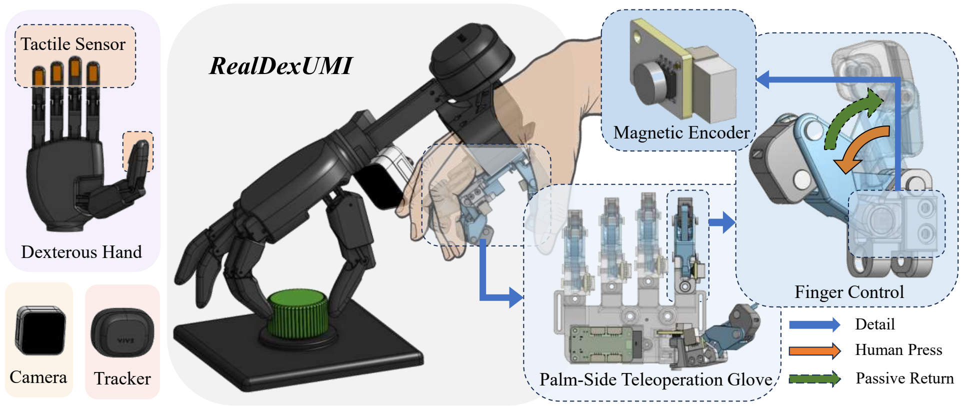


Fig. 2: **Hardware system.** The wearable interface combines a reusable dexterous end-effector module, a 6-DoF tracker, and a palm-side isomorphic teleoperation glove. The same hand and camera are used for wearable collection and robot deployment.

These streams form deployment-aligned trajectories: the visual and tactile observations are produced by the deployed sensing layout, and the hand-action labels are executable commands of the deployed hand rather than retargeted human-hand states. The policy observation is

$$o_t = \left(I_t, S_t^{\text{tactile}}, q_t^{\text{hand}}\right), \tag{1}$$

where I_t, S_t^{tactile}, and q_t^{hand} denote the in-hand image, tactile signal, and measured hand state.

Recording both measured states and executable commands preserves action–state correspondence under contact. The measured state reflects what contact allows, while the paired command preserves the operator's intended action. This provides supervision for maintaining and recovering contact, which is absent when measured hand states are used as action labels.

C. Hand-Frame Action and Deployment

The 6-DoF tracker provides the hand-frame pose $T_t \in SE(3)$ during collection. The policy does not observe this absolute pose; instead, we use it only to construct hand-frame relative motion labels:

$$\Delta T_{t,k} = T_t^{-1} T_{t+k}. \tag{2}$$

For each future step $t+k$, the action label is

$$a_{t,k} = \left[\Delta p_{t,k}, \Delta r_{t,k}, u_{t+k}^{\text{hand}}\right], \tag{3}$$

where $\Delta p_{t,k}$ and $\Delta r_{t,k}$ parameterize the relative end-effector motion, and u_{t+k}^{hand} is the executable hand command. The policy predicts a chunk of future hand-frame actions $\hat{A}_t = \pi_\theta(o_t)$. Because the motion label is expressed in the current hand frame, the policy learns local end-effector motion from local observations, making the action representation independent of the collection-specific global pose and robot base frame. At deployment, each robot realizes the predicted relative motion with its own IK and low-level controller, while the shared dexterous hand directly executes the predicted hand command. Thus, cross-embodiment deployment changes only the robot-side controller, not the learned policy or end-effector interface.

III. Experiments

We evaluate whether RealDexUMI demonstrations support deployable dexterous policy learning. Unless otherwise specified, policies are trained with ACT [43] using 200 RealDexUMI demonstrations per task and evaluated on a Franka FR3 equipped with the same dexterous end-effector module used during collection. Each policy is evaluated over 20 real-robot trials, and a trial is counted as successful only if the full task is completed. The task suite covers precision grasping, rotational manipulation, articulated-object interaction, tool use, precision insertion, multi-object grasping, long-horizon manipulation, and bimanual operation.

A. Real-World Dexterous Policy Learning

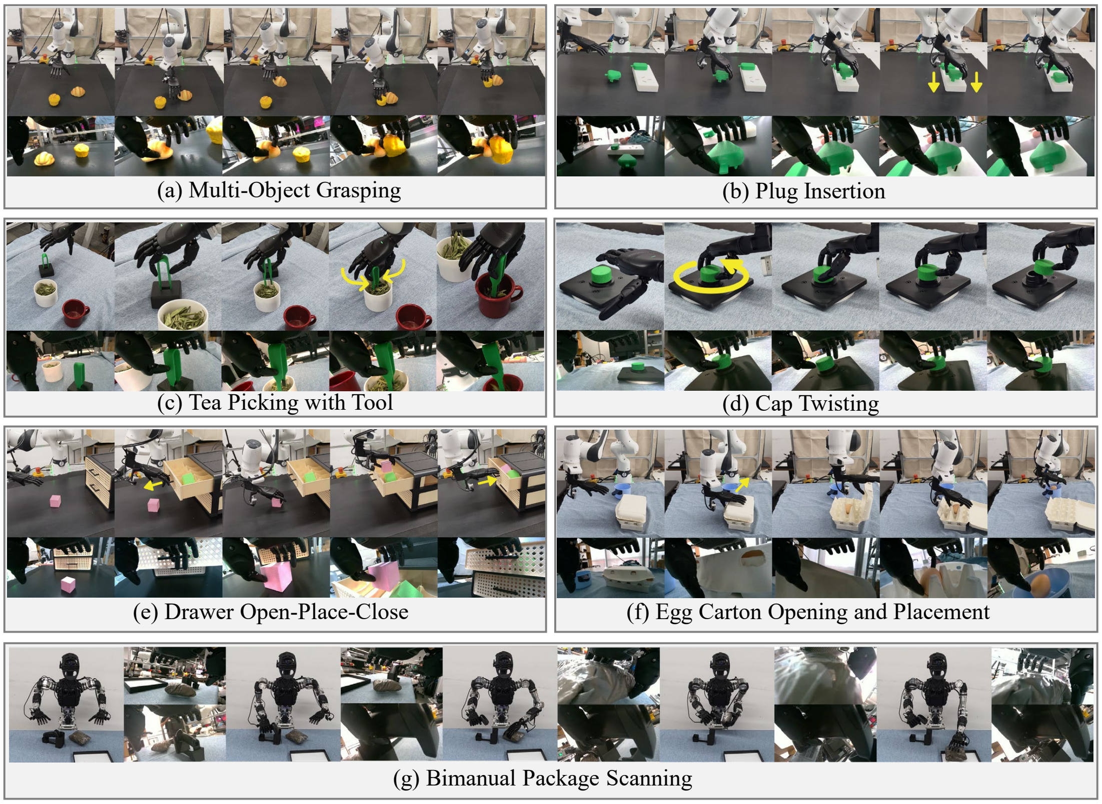
Table II reports full-task success across eight real-robot dexterous manipulation tasks, with representative rollouts shown in Fig. 3. RealDexUMI achieves an average success rate of 88.75%, covering fine-grained, contact-rich, long-horizon, and bimanual manipulation. For multi-stage tasks, we report final task completion in the main text; cumulative subgoal completion is provided in the appendix.

B. Ablations

We evaluate two single-factor ablations to test the importance of aligned sensing and executable hand-command labels. Removing tactile observations reduces average success from 88.75% to 70.00%, especially on tasks where contact is difficult to infer from vision alone, such as plug insertion and tea picking. Replacing executable hand-command labels with measured hand states further reduces success to 51.25%, supporting the importance of action–state correspondence for persistent contact and recovery. Together, these ablations indicate that RealDexUMI data is useful not only because the observations are aligned, but also because the action labels remain executable under contact.

C. Robustness and Cross-Embodiment Deployment

RealDexUMI policies also generalize beyond the nominal deployment setting. For cube pick-and-place, the same checkpoint succeeds in all 20 trials across unseen initial robot poses.



(a) Multi-Object Grasping (b) Plug Insertion (c) Tea Picking with Tool (d) Cap Twisting (e) Drawer Open-Place-Close (f) Egg Carton Opening and Placement (g) Bimanual Package Scanning

Fig. 3: **Policy rollouts.** Policies trained from RealDexUMI demonstrations execute representative tasks across multi-object grasping, precision insertion, tool use, twisting, articulated-object interaction, long-horizon execution, and bimanual operation.

TABLE II: Full-task success and single-factor ablations across eight real-robot tasks.

Method	**Cube**	**Multi-obj.**	**Plug**	**Cap**	**Tea**	**Drawer**	**Egg**	**Biman.**	**Avg.**
RealDexUMI	1.00	1.00	0.85	1.00	0.80	0.85	0.70	0.90	**88.75%**
w/o tactile	0.90	1.00	0.45	0.80	0.55	0.70	0.60	0.60	70.00%
State-as-action	0.65	0.85	0.30	0.35	0.20	0.45	0.60	0.70	51.25%

Note. Per-task entries report success rates in $[0, 1]$ over 20 trials, and Avg. reports the overall success percentage. The w/o tactile variant keeps executable hand-command labels but removes tactile observations. State-as-action keeps tactile observations but uses the next measured hand state as the hand-action label and executes the predicted state as the hand command.

We further deploy the same learned checkpoints across three heterogeneous robot embodiments by keeping the dexterous end-effector module and policy interface fixed while changing only the robot-side IK and low-level controller. As shown in Table III, the cube policy transfers without performance loss, while the drawer policy remains effective despite larger differences in arm kinematics and reachable workspace.

TABLE III: **Cross-embodiment success.**

Task	**FR3**	**RM65**	**Adam-U**
Cube	1.00	1.00	1.00
Drawer	0.85	0.75	0.80

Note. Entries report success rates over 20 trials.

IV. Discussion and Limitations

RealDexUMI suggests that dexterous robot learning benefits from demonstrations whose actions, contacts, and observations remain aligned with deployment. By using the same dexterous end effector for collection and execution, RealDexUMI provides deployment-aligned supervision for contact-rich manipulation, supporting skills such as precision insertion, tool use, twisting, long-horizon manipulation, and bimanual operation. The ablations further show that tactile sensing and executable hand-command labels are important when measured hand states alone do not capture the operator's intended correction.

The current system prioritizes local end-effector alignment, which limits tasks requiring object search, long-range planning, or explicit task-progress reasoning. Future global or egocentric views may address these limitations, but should preserve collection–deployment observation alignment. The current hand is designed to balance dexterity, weight, and wearable controllability. Its six actuated command DoFs keep the interface lightweight and intuitive, but provide less expressiveness than higher-DoF dexterous hands.

REFERENCES

[1] Xintao Chao, Shilong Mu, Yushan Liu, Shoujie Li, Chuqiao Lyu, Xiao-Ping Zhang, and Wenbo Ding. Exoviha: A cross-platform exoskeleton system with visual and haptic feedback for efficient dexterous skill learning. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 18732–18739. IEEE, 2025.

[2] Sirui Chen, Chen Wang, Kaden Nguyen, Li Fei-Fei, and C Karen Liu. Arcap: Collecting high-quality human demonstrations for robot learning with augmented reality feedback. *arXiv preprint arXiv:2410.08464*, 2024.

[3] Xitong Chen, Yifeng Pan, Min Li, and Xiaotian Ding. Dexvitac: Collecting human visuo-tactile-kinematic demonstrations for contact-rich dexterous manipulation. *arXiv preprint arXiv:2603.17851*, 2026.

[4] Tailai Cheng, Kejia Chen, Lingyun Chen, Liding Zhang, Yue Zhang, Yao Ling, Mahdi Hamad, Zhenshan Bing, Fan Wu, Karan Sharma, et al. Tacumi: A multi-modal universal manipulation interface for contact-rich tasks. *arXiv preprint arXiv:2601.14550*, 2026.

[5] Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-television: Teleoperation with immersive active visual feedback. *arXiv preprint arXiv:2407.01512*, 2024.

[6] 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 *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[7] Runyu Ding, Yuzhe Qin, Jiyue Zhu, Chengzhe Jia, Shiqi Yang, Ruihan Yang, Xiaojuan Qi, and Xiaolong Wang. Bunny-visionpro: Real-time bimanual dexterous teleoperation for imitation learning. 2024. URL https://arxiv.org/abs/2407.03162.

[8] Jinda Du, Jieji Ren, Qiaojun Yu, Ningbin Zhang, Yu Deng, Xingyu Wei, Yufei Liu, Guoying Gu, and Xiangyang Zhu. Mile: A mechanically isomorphic exoskeleton data collection system with fingertip visuotactile sensing for dexterous manipulation. *arXiv preprint arXiv:2512.00324*, 2025.

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

[10] Hongjie Fang, Chenxi Wang, Yiming Wang, Jingjing Chen, Shangning Xia, Jun Lv, Zihao He, Xiyan Yi, Yunhan Guo, Xinyu Zhan, Lixin Yang, Weiming Wang, Cewu Lu, and Hao-Shu Fang. Airexo-2: Scaling up generalizable robotic imitation learning with low-cost exoskeletons. *arXiv preprint arXiv:03081*, 2025.

[11] Yuyang Gao, Haofei Ma, and Pai Zheng. Glovity: Learning dexterous contact-rich manipulation via spatial wrench feedback teleoperation system. *arXiv preprint arXiv:2510.09229*, 2025.

[12] Huy Ha, Yihuai Gao, Zipeng Fu, Jie Tan, and Shuran Song. UMI on legs: Making manipulation policies mobile with manipulation-centric whole-body controllers. In *Proceedings of the 2024 Conference on Robot Learning*, 2024.

[13] Binghao Huang, Yixuan Wang, Xinyi Yang, Yiyue Luo, and Yunzhu Li. 3d-vitac: Learning fine-grained manipulation with visuo-tactile sensing. *arXiv preprint arXiv:2410.24091*, 2024.

[14] Haoran Huang, Haonan Dong, and Huixu Dong. Mobile umi: Cross-view diffusion policy with decoupled kinematics for mobile manipulation. *arXiv preprint arXiv:2605.20894*, 2026.

[15] Aadhithya Iyer, Zhuoran Peng, Yinlong Dai, Irmak Guzey, Siddhant Haldar, Soumith Chintala, and Lerrel Pinto. Open teach: A versatile teleoperation system for robotic manipulation. *arXiv preprint arXiv:2403.07870*, 2024.

[16] Feiyu Jia, Xiaojie Niu, Sizhe Yang, Qingwei Ben, Tao Huang, Jingbo Wang, Jiangmiao Pang, et al. Feel robot feels: Tactile feedback array glove for dexterous manipulation. *arXiv preprint arXiv:2603.28542*, 2026.

[17] Joonho Koh, Haechan Jung, Nayoung Kim, Wook Ko, and Changjoo Nam. Dex-mouse: A low-cost portable and universal interface with force feedback for data collection of dexterous robotic hands. *arXiv preprint arXiv:2604.15013*, 2026.

[18] Huayue Liang, Ruochong Li, Yaodong Yang, Long Zeng, Yuanpei Chen, and Xueqian Wang. Cdf-glove: A cable-driven force feedback glove for dexterous teleoperation. *arXiv preprint arXiv:2603.05804*, 2026.

[19] Fanqi Lin, Yingdong Hu, Pingyue Sheng, Chuan Wen, Jiacheng You, and Yang Gao. Data scaling laws in imitation learning for robotic manipulation. In *International Conference on Learning Representations*, volume 2025, pages 54877–54910, 2025.

[20] Fangchen Liu, Chuanyu Li, Yihua Qin, Jing Xu, Pieter Abbeel, and Rui Chen. Vitamin: Learning contact-rich tasks through robot-free visuo-tactile manipulation interface. *arXiv preprint arXiv:2504.06156*, 2025.

[21] Kehui Liu, Chuyue Guan, Zhongjie Jia, Ziniu Wu, Xin Liu, Tianyu Wang, Shuai Liang, Pengan Chen, Pingrui Zhang, Haoming Song, et al. Fastumi: A scalable and hardware-independent universal manipulation interface with dataset. *arXiv preprint arXiv:2409.19499*, 2024.

[22] Kehui Liu, Zhongjie Jia, Yang Li, Pengan Chen, Song Liu, Xin Liu, Pingrui Zhang, Haoming Song, Xinyi Ye, Nieqing Cao, et al. Fastumi-100k: Advancing data-driven robotic manipulation with a large-scale umi-style dataset. *arXiv preprint arXiv:2510.08022*, 2025.

[23] Zeyi Liu, Cheng Chi, Eric Cousineau, Naveen Kuppuswamy, Benjamin Burchfiel, and Shuran Song. Maniwav: Learning robot manipulation from in-the-wild audio-visual data. *arXiv preprint arXiv:2406.19464*,

2024.
[24] Shaqi Luo, Yuanyuan Li, Youhao Hu, Chenhao Yu, Chaoran Xu, Jiachen Zhang, Guocai Yao, Tiejun Huang, Ran He, and Zhongyuan Wang. Omniumi: Towards physically grounded robot learning via human-aligned multimodal interaction. *arXiv preprint arXiv:2604.10647*, 2026.
[25] Younghyo Park and Pulkit Agrawal. Using apple vision pro to train and control robots, 2024. URL https://github.com/Improbable-AI/VisionProTeleop.
[26] 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*, 2023.
[27] Branden Romero, Hao-Shu Fang, Pulkit Agrawal, and Edward Adelson. Eyesight hand: Design of a fully-actuated dexterous robot hand with integrated vision-based tactile sensors and compliant actuation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1853–1860. IEEE, 2024.
[28] Mingyo Seo, H. Andy Park, Shenli Yuan, Yuke Zhu, and Luis Sentis. Legato: Cross-embodiment imitation using a grasping tool. *IEEE Robotics and Automation Letters (RA-L)*, 2025.
[29] Zilin Si, Jose Enrique Chen, M. Emre Karagozler, Antonia Bronars, Jonathan Hutchinson, Thomas Lampe, Nimrod Gileadi, Taylor Howell, Stefano Saliceti, Lukasz Barczyk, Ilan Olivarez Correa, Tom Erez, Mohit Shridhar, Murilo Fernandes Martins, Konstantinos Bousmalis, Nicolas Heess, Francesco Nori, and Maria Bauza. Exostart: Efficient learning for dexterous manipulation with sensorized exoskeleton demonstrations, 2025. URL https://arxiv.org/abs/2506.11775.
[30] Tony Tao, Mohan Kumar Srirama, Jason Jingzhou Liu, Kenneth Shaw, and Deepak Pathak. Dexwild: Dexterous human interactions for in-the-wild robot policies. *arXiv preprint arXiv:2505.07813*, 2025.
[31] 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. *arXiv preprint arXiv:2403.07788*, 2024.
[32] Junming Wang, Teng Pu, Wingmun Fung, Jindong Wang, Shanchang Wang, Yuan Deng, Shuyuan Wang, Ziwei Liu, Kunhao Pan, Ping Yang, et al. Xrzero-g0: Pushing the frontier of dexterous robotic manipulation with interfaces, quality and ratios. *arXiv preprint arXiv:2604.13001*, 2026.
[33] Dehao Wei and Huazhe Xu. A wearable robotic hand for hand-over-hand imitation learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 18113–18119. IEEE, 2024.
[34] Shihan Wu, Xuecheng Liu, Shaoxuan Xie, Pengwei Wang, Xinghang Li, Bowen Yang, Zhe Li, Kai Zhu, Hongyu Wu, Yiheng Liu, et al. Robocoin: An open-sourced bimanual robotic data collection for integrated manipulation. *arXiv preprint arXiv:2511.17441*, 2025.
[35] Chendong Xin, Mingrui Yu, Yongpeng Jiang, Zhefeng Zhang, and Xiang Li. Analyzing key objectives in human-to-robot retargeting for dexterous manipulation. *IEEE Robotics and Automation Practice*, 2026.
[36] Mengda Xu, Han Zhang, Yifan Hou, Zhenjia Xu, Linxi Fan, Manuela Veloso, and Shuran Song. Dexumi: Using human hand as the universal manipulation interface for dexterous manipulation. In *Conference on Robot Learning*, pages 437–459. PMLR, 2025.
[37] Xiaomeng Xu, Jisang Park, Han Zhang, Eric Cousineau, Aditya Bhat, Jose Barreiros, Dian Wang, and Shuran Song. Hommi: Learning whole-body mobile manipulation from human demonstrations. *arXiv preprint arXiv:2603.03243*, 2026.
[38] Yue Xu, Litao Wei, Pengyu An, Qingyu Zhang, and Yong-Lu Li. exumi: Extensible robot teaching system with action-aware task-agnostic tactile representation. *arXiv preprint arXiv:2509.14688*, 2025.
[39] Shiqi Yang, Minghuan Liu, Yuzhe Qin, Runyu Ding, Jialong Li, Xuxin Cheng, Ruihan Yang, Sha Yi, and Xiaolong Wang. Ace: A cross-platform visual-exoskeletons system for low-cost dexterous teleoperation. *arXiv preprint arXiv:2408.11805*, 2024.
[40] Chenhao Yu, Hongwu Wang, Youhao Hu, Jiachen Zhang, Yuanyuan Li, and Shaqi Luo. Bifrostumi: Bridging robot-free demonstrations and humanoid whole-body manipulation. *arXiv preprint arXiv:2605.03452*, 2026.
[41] Qiyuan Zeng, Chengmeng Li, Jude St John, Zhongyi Zhou, Junjie Wen, Guorui Feng, Yichen Zhu, and Yi Xu. Activeumi: Robotic manipulation with active perception from robot-free human demonstrations. *arXiv preprint arXiv:2510.01607*, 2025.
[42] Han Zhang, Songbo Hu, Zhecheng Yuan, and Huazhe Xu. Doglove: Dexterous manipulation with a low-cost open-source haptic force feedback glove. *arXiv preprint arXiv:2502.07730*, 2025.
[43] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*, 2023. URL https://openreview.net/forum?id=e8Eu1lqLaf.
[44] Yupeng Zheng, Jichao Peng, Weize Li, Yuhang Zheng, Xiang Li, Yujie Jin, Julong Wei, Guanhua Zhang, Ruiling Zheng, Ming Cao, et al. World in your hands: A large-scale and open-source ecosystem for learning human-centric manipulation in the wild. *arXiv preprint arXiv:2512.24310*, 2025.
[45] Alvin Zhu, Mingzhang Zhu, Beom Jun Kim, Jose Victor SH Ramos, Yike Shi, Yufeng Wu, Raayan Dhar, Fuyi Yang, Ruochen Hou, Hanzhang Fang, et al. Dexexo: A wearability-first dexterous exoskeleton for operator-agnostic demonstration and learning. *arXiv preprint arXiv:2603.17323*, 2026.
[46] Xinyue Zhu, Binghao Huang, and Yunzhu Li. Touch in the wild: Learning fine-grained manipulation with a portable visuo-tactile gripper. In *The Thirty-ninth*

Annual Conference on Neural Information Processing Systems, 2025. URL https://openreview.net/forum?id=WabVVQKTUF.

APPENDIX A
HARDWARE AND EMBEDDED INTERFACE

This section provides implementation details of the palm-side teleoperation glove used for RealDexUMI data collection. The glove measures operator inputs and converts them into executable hand commands in the deployed hand's command space. Fig. 4 shows the overall embedded interface.

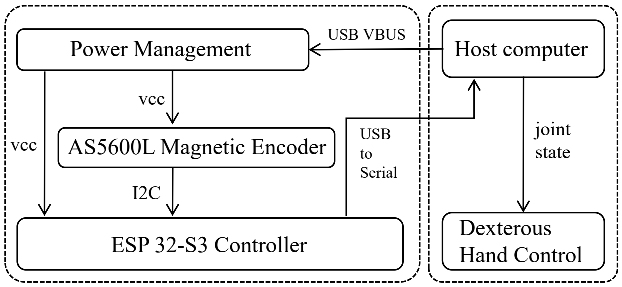


Fig. 4: **Glove embedded interface.** Six AS5600L magnetic encoders measure the actuated glove DoFs. The ESP32-S3 controller reads encoder values through I2C and streams the resulting 6-D command vector to the host computer through USB serial.

A. Glove Sensing Interface

The glove measures six actuated command DoFs using magnetic encoders. Each measured DoF corresponds to one actuated DoF of the RealDexUMI hand, producing a 6-D command vector in the deployed hand's command space. Unlike human-hand motion capture, this interface does not estimate fingertip poses or solve a retargeting problem. It directly records the command variables later executed by the robot hand.

Fig. 5 illustrates the single-joint encoder design. For each sensed DoF, a diametric magnet is aligned with the joint rotation axis and placed above an AS5600L sensor. The sensor provides an absolute angular reading by measuring the magnetic field direction of the rotating magnet. This non-contact measurement avoids mechanical friction in the sensing path and allows the glove to recover the current joint reading after power-on.

B. Embedded Controller

An ESP32-S3 controller reads the six encoder values through the I2C bus and streams the measurements to the host computer through USB serial, as shown in Fig. 4. The controller packages the encoder readings into a fixed-order joint vector, which is then used by the host-side teleoperation process to command the dexterous hand. This design keeps the glove self-contained and allows it to be used without an external data-acquisition board.

C. Command Calibration and Mapping

Before each collection session, the glove is held at a reference open pose to record encoder offsets. During operation, each encoder reading is converted into a relative displacement from this reference pose and linearly mapped to the corresponding hand command range. The resulting command vector is recorded as the hand action label in the demonstration dataset and sent to the hand during teleoperation.

TABLE IV: Recorded streams in RealDexUMI demonstrations.

Stream	Rate	Observation	Action
In-hand RGB	30 Hz	Yes	No
Fingertip tactile	20 Hz	Yes	No
Hand state	100 Hz	Yes	No
Glove command	200 Hz	No	Yes
End-effector pose	100 Hz	No	Yes

APPENDIX B
DATA COLLECTION AND ACTION–STATE CORRESPONDENCE

RealDexUMI records all demonstration streams at the end effector, including in-hand RGB observations, fingertip tactile signals, measured hand states, glove commands, and 6-DoF end-effector poses. The RGB, tactile, hand-state, and hand-command streams are generated by the same dexterous end-effector module used during robot deployment. The 6-DoF pose stream is used to construct relative end-effector action labels, but is not included in the policy observation.

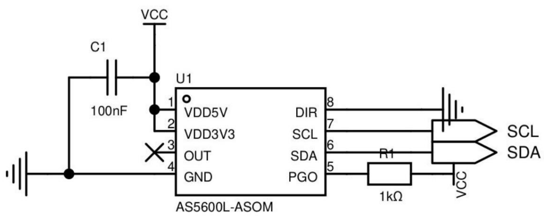
A. Recorded Streams

Table IV summarizes the recorded streams and their use in policy learning. The policy observation consists of in-hand RGB, tactile signals, and measured hand states. The action label consists of hand-frame relative end-effector motion and executable hand commands.

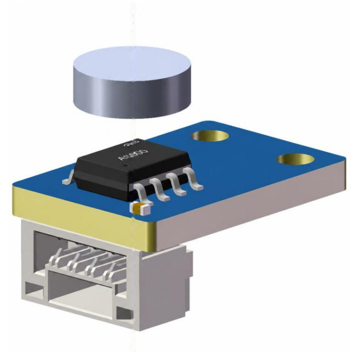
B. Latest-Sample Synchronization

The recorded streams run at different rates, while policy training uses fixed-rate demonstration steps. For each policy timestep, we use the in-hand RGB timestamp as the temporal anchor. For every non-visual stream, we select the latest available sample whose timestamp is not later than the anchor timestamp. This latest-sample strategy avoids extrapolating sensor values and keeps each training sample grounded in measurements that were already available at that time.

The same synchronized timestep provides both policy observations and action labels. The observation is formed from the RGB frame, the latest tactile signal, and the latest measured hand state. For each future action step $t+k$, the hand-command label is selected from the latest glove command not later than the corresponding future timestamp. The end-effector motion label is computed from the synchronized 6-DoF pose sequence as a relative transform in the current hand frame. Thus, the policy learns from local end-effector observations and predicts actions in the same hand-centric interface used during deployment.



(a) AS5600L interface circuit



(b) Magnetic joint encoder

Fig. 5: **Single-joint magnetic encoder design.** (a) AS5600L interface circuit. The encoder communicates with the controller through the I2C bus using SCL and SDA. (b) Magnetic joint encoder. A diametric magnet is aligned with the joint rotation axis and placed above the AS5600L sensor for absolute angle measurement.

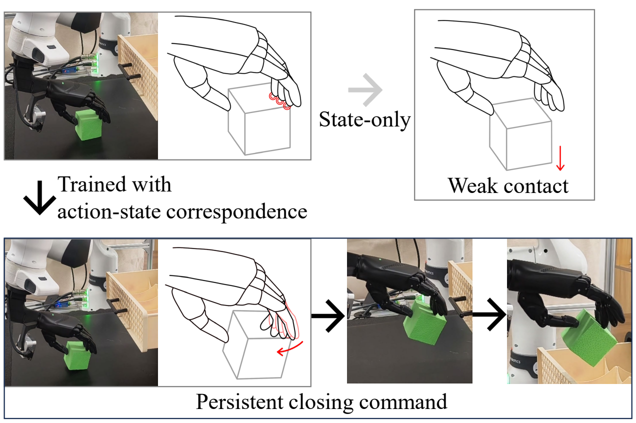


Fig. 6: **Action–state correspondence.** Paired executable hand actions and measured hand states provide supervision for contact-aware corrections in contact-rich manipulation, which state-only supervision cannot provide.

C. Action–State Correspondence

RealDexUMI records both the measured hand state and the executable hand command. This distinction is important under contact. The measured state reflects the posture the hand reaches after object contact constrains finger motion, while the executable command preserves the operator's intended correction, such as continuing to close, pinch, twist, or stabilize. Fig. 6 illustrates this action–state correspondence.

Appendix C
Task Definitions and Evaluation Protocol

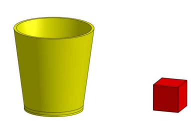
The task suite covers basic grasping, rotational manipulation, articulated-object interaction, tool use, multi-stage manipulation, precision insertion, multi-object grasping, and bimanual coordination. Each policy is evaluated over 20 real-robot trials per task. A trial is counted as successful only if the final task goal is completed within the task timeout. Table V summarizes the full-task success criterion for each task.

A. Evaluation Protocol

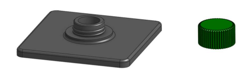
For each task, the initial object and robot configurations are varied within the feasible workspace of the robot. All trials are evaluated on physical robot rollouts without resetting the policy during execution. For long-horizon tasks, intermediate subgoals are recorded in the order required by the task. A later subgoal is counted only when all preceding subgoals in the same rollout have been completed.

This cumulative subgoal protocol is used only to diagnose failure modes. We do not average subgoal success rates as independent trials, because later subgoals are conditioned on earlier task progress. The final task success reported in the main paper is therefore the primary metric.

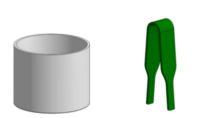
B. Representative Task Props



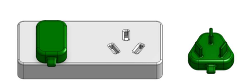
(a) Cube pick-and-place.



(b) Cap twisting.



(c) Tea picking with tool.



(d) Plug insertion.

Fig. 7: **Representative task props.** We use task-specific fixtures and props to standardize task geometry when needed, including a cup-and-cube setup for cube pick-and-place, a ridged cap fixture for cap twisting, a tweezer-and-cup setup for tea picking, and a plug fixture for insertion.

Several tasks use simple fixtures or 3D-printed props to standardize task geometry and improve reproducibility. These props are used where controlled geometry is useful, but not

TABLE V: Task definitions and full-task success criteria.

Task	Full-task success criterion
Cube pick-and-place	The cube is placed into the target cup.
Cap twisting	The cap is twisted off from the fixture.
Drawer stowing	The drawer is opened, the cube is placed inside, and the drawer is closed.
Tea picking with tool	The tweezers are grasped and used to transfer tea into the target cup.
Egg carton handling	The carton is opened, the egg is picked, and the egg is placed into the pot.
Plug insertion	The plug is inserted into the target socket.
Multi-object grasping	Both target objects are grasped in sequence.
Bimanual package scanning	The package is manipulated bimanually and scanned successfully.

every task requires a printed asset. Fig. 7 shows representative props used in the cube pick-and-place, cap twisting, tea picking, and plug insertion tasks.

APPENDIX D
TRAINING DETAILS

We train ACT for the main real-robot policy experiments and evaluate Diffusion Policy as an additional imitation-learning baseline. Both policy families use the same RealDex-UMI observation and action interface. The observation consists of in-hand RGB, fingertip tactile signals, and measured hand states. The action consists of hand-frame relative end-effector motion and executable hand commands. Table VI summarizes the training hyperparameters used for ACT and Diffusion Policy. Fig. 11 reports the action-prediction L1 loss during ACT training. We include this plot as an optimization diagnostic; real-robot success rates remain the primary evaluation metric.

A. Diffusion Policy Baseline

We additionally train Diffusion Policy on the same RealDexUMI demonstrations using the same observation and action interface as the ACT policies in the main experiments. This evaluation tests whether RealDexUMI data can support another common imitation-learning policy backend. Table VII reports the Diffusion Policy results across the same eight real-robot tasks.

In our implementation, Diffusion Policy underperforms ACT on several precision- and contact-rich tasks. We use this experiment to test backend compatibility rather than to exhaustively tune policy architectures, and therefore use ACT as the primary backend in the main experiments.

APPENDIX E
ADDITIONAL EVALUATION RESULTS

A. Initial-Pose Robustness

We evaluate whether the learned cube pick-and-place policy remains robust to initial robot poses outside the demonstration distribution. Fig. 8 shows representative evaluation settings. The same checkpoint is used across all initial-pose trials without retraining or task-specific adaptation.

B. Cross-Embodiment Deployment

We further evaluate whether the same learned policies can deploy across heterogeneous robot embodiments. Fig. 9 shows cross-embodiment deployment of the drawer-stowing policy on Franka FR3, RealMan RM65, and PND Adam-U. The policy checkpoint and dexterous end-effector module are kept fixed; only the robot-side IK and low-level controller are changed.

C. Cumulative Subgoal Completion

For multi-stage tasks, we report cumulative subgoal completion to show where failures occur. A later subgoal is counted only if all preceding subgoals in the same rollout have been completed. Therefore, these values are not averaged as independent subgoal success rates; the final full-task success remains the primary metric reported in the main paper. Table VIII reports cumulative subgoal completion for the multi-stage tasks.

APPENDIX F
COLLECTION-TIME DEXTERITY

We evaluate collection-time control on two dexterous tasks: cap twisting and tea picking with tweezers. We compare RealDexUMI with AVP-based arm–hand teleoperation [25] and Manus-glove retargeting to the RealDexUMI hand [35], and include direct human-hand manipulation as a reference. The same operator performs all methods and receives 10 minutes of practice for each interface before evaluation. Each method is evaluated over 10 trials per task. For AVP, wrist and hand information are captured and retargeted to the Franka arm and RealDexUMI end-effector module. For Manus-glove retargeting, the glove measurements are mapped to the RealDexUMI hand for collection-time control. Completion time is averaged over successful trials, and trials exceeding 5 min are counted as failures.

As shown in Fig. 10, success rate is the primary metric, since completion time is averaged only over successful trials and therefore measures efficiency conditional on task completion. RealDexUMI achieves the highest success rate and shortest completion time among deployable collection interfaces.

The Manus baseline succeeds on cap twisting, suggesting that wearable hand input can make wrist-driven teleoperation intuitive. However, its performance drops sharply on tea picking, where tweezer use depends on precise fingertip pinching. This contrast shows that rich human-hand motion alone is insufficient when it must be retargeted to a robot hand. RealDexUMI instead provides precise linear control directly in the dexterous hand's action space, enabling reliable deployable demonstrations.

APPENDIX G
ADDITIONAL SYSTEM COMPARISON

To support the perceived-complexity label in Table I, we conduct a lightweight structured survey using available system descriptions, device images, and demonstration materials. The rating reflects perceived setup and operation complexity and should be interpreted as a coarse comparative indicator rather than a direct hands-on usability measurement.

Evaluators rate each system on a three-level scale: Low, Medium, or High. The final label is determined by majority vote across 20 evaluators with backgrounds in robotics or related engineering fields. Fig. 12 shows the survey form provided to evaluators. Table IX reports the vote counts and final perceived-complexity labels.

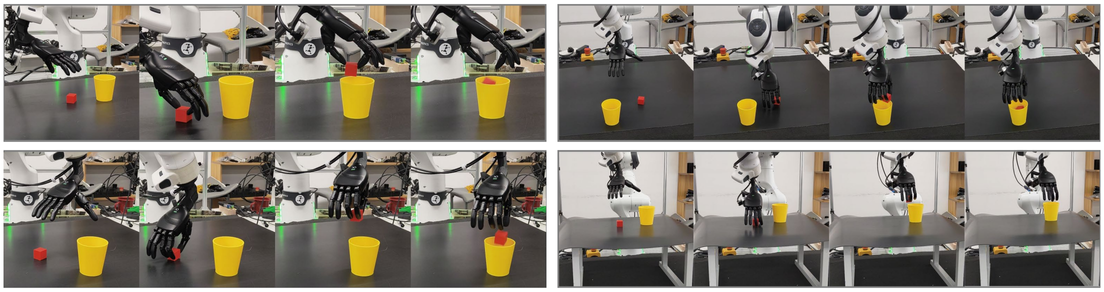


Fig. 8: **Initial-pose robustness.** A cube pick-and-place policy is evaluated under unseen initial robot poses outside the demonstration distribution.

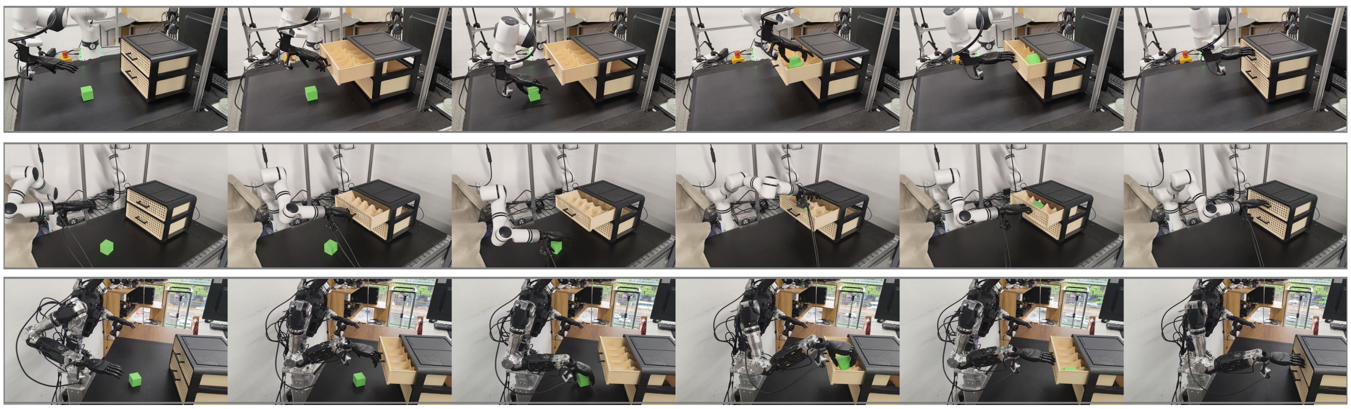


Fig. 9: **Cross-embodiment deployment.** The same drawer-stowing checkpoint runs on Franka FR3, RealMan RM65, and PND Adam-U without retraining.

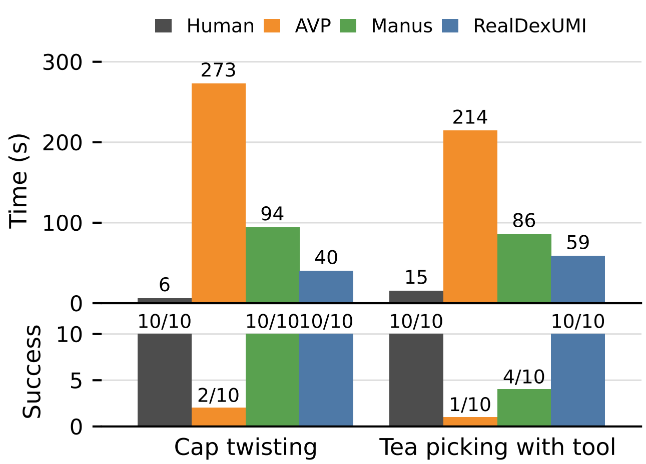


Fig. 10: **Teleoperation comparison.** Time is averaged over successful trials. Trials exceeding 5 min are counted as failures.

TABLE VI: Training hyperparameters for ACT and Diffusion Policy.

Group	**Parameter**	**ACT**	**Diffusion Policy**
Observation	Image obs. horizon	1	2
	Proprio. obs. horizon	1	2
	Action horizon	20	32
	Executed action steps	20	32
	Obs. resolution	256×256	256×256
Architecture	Vision encoder	ResNet-18	ResNet-18
	Pretrained vision weights	ImageNet	None
	Transformer dim.	512	–
	Transformer layers	4 encoder, 1 decoder	–
	Attention heads	8	–
	ACT latent dim.	32	–
	KL weight	10	–
	U-Net channels	–	$(512, 1024, 2048)$
Optimization	Optimizer	AdamW	Adam
	Learning rate	1×10^{-5}	1×10^{-4}
	Momentum	$\beta_1 = 0.9,\ \beta_2 = 0.999$	$\beta_1 = 0.95,\ \beta_2 = 0.999$
	Weight decay	1×10^{-4}	1×10^{-6}
	LR schedule	None	Cosine, 500 warmup steps
	Batch size	128	128
	Training steps	200k	100k
	Grad. clip norm	10	10
Diffusion	Train diffusion steps	–	100
	Inference steps	–	100
	Noise scheduler	–	DDPM
	Beta schedule	–	Squared cosine
	Prediction target	–	Noise ϵ

TABLE VII: Diffusion Policy success rates across eight real-robot tasks.

Method	**Cube**	**Multi-obj.**	**Plug**	**Cap**	**Tea**	**Drawer**	**Egg**	**Biman.**	**Avg.**
Diffusion Policy	0.85	0.75	0.25	0.75	0.40	0.60	0.80	0.70	63.75%

Note. Per-task entries report success rates in $[0, 1]$ over 20 trials, and Avg. reports the overall success percentage.

TABLE VIII: Cumulative subgoal completion for multi-stage tasks.

Task	**Step**	**Subgoal**	**Cumulative completion**
Drawer stowing	1	Open drawer	20/20
	2	Pick cube	18/20
	3	Close drawer	17/20
Tea picking with tool	1	Grasp tweezers	17/20
	2	Pick tea	16/20
	3	Place tea in cup	16/20
Egg carton handling	1	Open carton	15/20
	2	Pick egg	14/20
	3	Place egg in pot	14/20
Multi-object grasping	1	Grasp first object	20/20
	2	Grasp second object	20/20

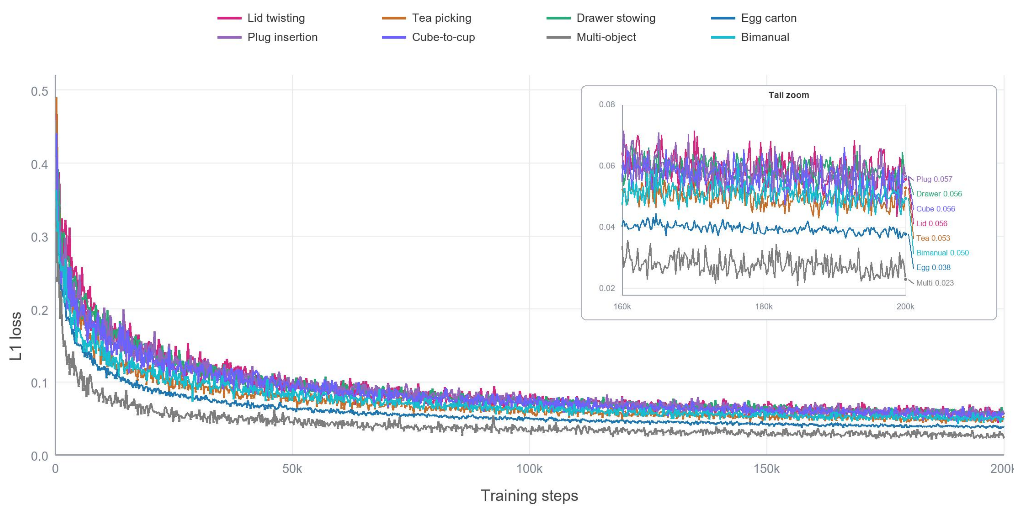
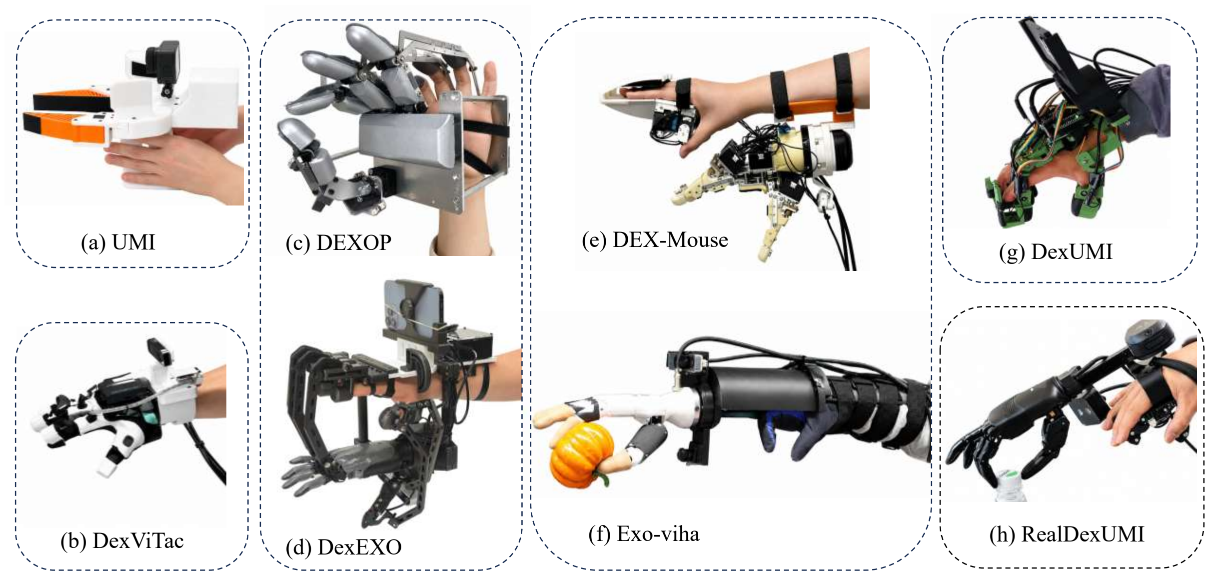


Fig. 11: **Policy training loss.** Action-prediction L1 loss during ACT training.



Survey question: How complex does the system appear for teleoperation-based demonstration collection, considering wearing/holding, setup, adjustment, and operation?

System	**Low**	**Medium**	**High**
(a) UMI [6]	☐	☐	☐
(b) DexViTac [3]	☐	☐	☐
(c) DEXOP [9]	☐	☐	☐
(d) DexExo [45]	☐	☐	☐
(e) DEX-Mouse [17]	☐	☐	☐
(f) Exo-ViHa [1]	☐	☐	☐
(g) DexUMI [36]	☐	☐	☐
(h) RealDexUMI	☐	☐	☐

Fig. 12: **Survey form for perceived teleoperation complexity.** Evaluators rate the perceived setup and operation complexity of each demonstration interface using a three-level scale.

TABLE IX: Survey results for perceived teleoperation complexity. Each entry reports the number of votes among 20 evaluators.

System	**Low**	**Med.**	**High**	**Final**
UMI [6]	18	2	0	Low
DexViTac [3]	15	5	0	Low
DEXOP [9]	0	0	20	High
DexExo [45]	1	6	13	High
DEX-Mouse [17]	1	5	14	High
Exo-ViHa [1]	0	1	19	High
DexUMI [36]	6	10	4	Medium
RealDexUMI	16	4	0	Low