AirExo-2: SCALING UP GENERALIZABLE ROBOTIC IMITATION LEARNING WITH LOW-COST EXOSKELETONS

Hongjie Fang[*,1], **Chenxi Wang**[*,2], **Yiming Wang**[*,1], **Jingjing Chen**[*,1],
Shangning Xia[1], **Jun Lv**[1,2], **Zihao He**[1], **Xiyan Yi**[1], **Yunhan Guo**[1], **Xinyu Zhan**[1],
Lixin Yang[1], **Weiming Wang**[1], **Cewu Lu**[1,2,†], **Hao-Shu Fang**[1,†]
[1]Shanghai Jiao Tong University, [2]Shanghai Noematrix Intelligence Technology Ltd.
[*]Equal Contribution, [†]Corresponding Authors

ABSTRACT

Scaling up imitation learning for real-world applications requires efficient and cost-effective demonstration collection methods. Current teleoperation approaches, though effective, are expensive and inefficient due to the dependency on physical robot platforms. Alternative data sources like in-the-wild demonstrations can eliminate the need for physical robots and offer more scalable solutions. However, existing in-the-wild data collection devices have limitations: handheld devices offer restricted in-hand camera observation, while whole-body devices often require fine-tuning with robot data due to action inaccuracies. In this paper, we propose *AirExo*-2, a low-cost exoskeleton system for large-scale in-the-wild demonstration collection. By introducing the demonstration adaptor to transform the collected in-the-wild demonstrations into pseudo-robot demonstrations, our system addresses key challenges in utilizing in-the-wild demonstrations for downstream imitation learning in real-world environments. Additionally, we present *RISE*-2, a generalizable policy that integrates 2D and 3D perceptions, outperforming previous imitation learning policies in both in-domain and out-of-domain tasks, even with limited demonstrations. By leveraging in-the-wild demonstrations collected and transformed by the *AirExo*-2 system, without the need for additional robot demonstrations, *RISE*-2 achieves comparable or superior performance to policies trained with teleoperated data, highlighting the potential of *AirExo*-2 for scalable and generalizable imitation learning. Project website: https://airexo.tech/airexo2/.

1 INTRODUCTION

Scaling up generalizable robotic imitation learning in the real world is essential for developing robust policies that can be directly applied to practical scenarios (Bharadhwaj, 2024; Black et al., 2024). While teleoperation has been commonly used to collect demonstrations for imitation learning, it requires a physical robot platform to record both observations and robot actions, raising costs due to the expensive hardware involved. Despite its effectiveness, this approach is costly and inefficient for scaling up demonstration collection for imitation learning.

Recently, researchers have explored several alternative data sources that can be scaled up at a lower cost, such as human videos (Bharadhwaj et al., 2024a; Smith et al., 2020; Vosylius & Johns, 2024; Wang et al., 2023; Wen et al., 2023; Xiong et al., 2021) and in-the-wild demonstrations (Fang et al., 2024b; Chi et al., 2024; Shafiullah et al., 2023; Wang et al., 2024a; Young et al., 2020). Unlike traditional robot-centric demonstrations collected through teleoperation, both sources focus on human-centric demonstrations. This eliminates the need for a physical robot during data collection, greatly reducing costs and enhancing scalability. For human videos, actions are typically inferred using hand detection and pose estimation (Papagiannis et al., 2024; Vosylius & Johns, 2024), or inferred from object pose during interactions (Heppert et al., 2024; Xu et al., 2024). In contrast, in-the-wild demonstrations utilize additional devices to interface humans and robots, capturing the observation and action data relevant to robotic manipulations when the human is performing the task.

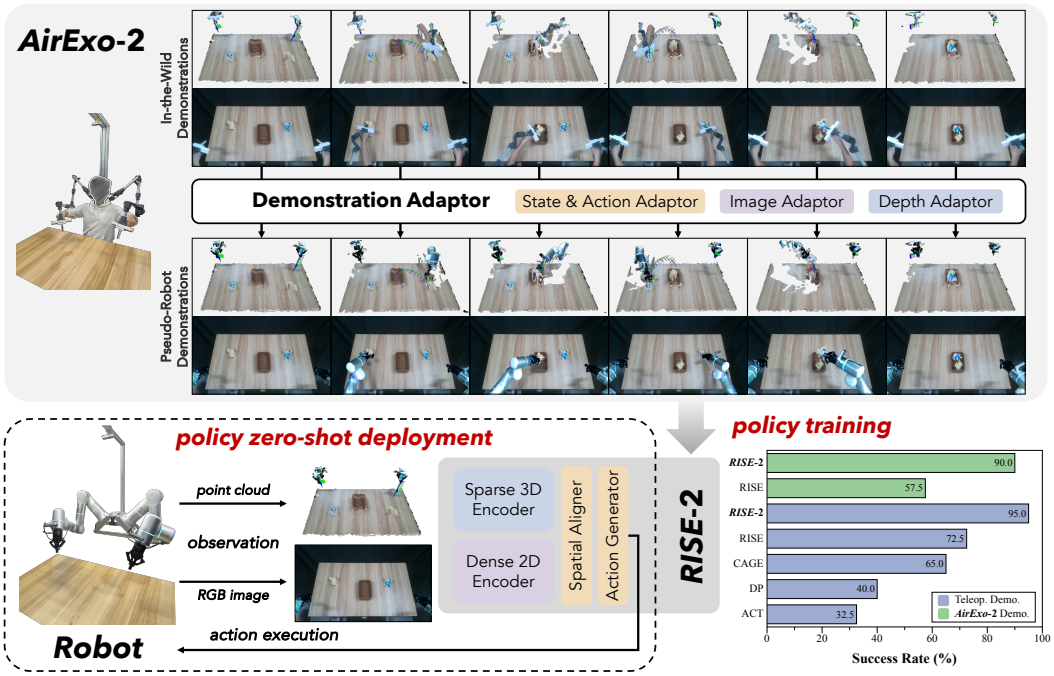


Figure 1: Overview of the *AirExo*-2 System and the *RISE*-2 Policy. (Top) The ***AirExo*-2** system enables the scalable collection and effective adaptation of in-the-wild demonstration data. A demonstration adaptor is employed to convert in-the-wild demonstrations into pseudo-robot demonstrations that are directly usable for training imitation learning policies. (Bottom) The proposed generalizable policy, ***RISE*-2**, can effectively leverage these converted pseudo-robot demonstrations for learning manipulation skills, enabling zero-shot deployment on real-world dual-arm robots without requiring any teleoperated demonstrations, and achieving results comparable to policies trained with the same amount of teleoperated data.

Current devices for collecting in-the-wild demonstrations can be broadly categorized into two classes: handheld devices (Chi et al., 2024; Etukuru et al., 2024; Shafiullah et al., 2023; Young et al., 2020) and whole-body devices (Fang et al., 2024b; Chen et al., 2024c; Kim et al., 2023a). Handheld devices are typically designed with end-effectors identical to those of the robot. By equipping both the handheld devices and the robot with in-hand cameras, these methods leverage in-hand observations to ensure visual consistency. However, they have two main limitations: (1) the pose estimation of the devices often relies on visual SLAM algorithms, which can introduce inaccuracies in capturing actions (§6.2), and (2) the in-hand camera has a limited field of view and struggles to capture accurate depth information during interactions between the robot and objects (§5.2). On the contrary, whole-body devices are usually more accurate in action capturing and offer flexible observation options. However, they generally use in-the-wild demonstrations for pre-training and require additional teleoperation data to fine-tune the policy. The underlying reason for this limitation is that *the data from in-the-wild demonstrations still exhibits a domain gap compared to the data from the real robot*.

To address these challenges, we introduce the ***AirExo*-2** system for large-scale in-the-wild demonstration collection and adaptation. As shown at the top of Fig. 1, we propose a demonstration adaptor to transform the in-the-wild demonstrations into pseudo-robot demonstrations, both observations and actions from the in-the-wild demonstrations can be effectively aligned with the robot domain. This transformation enables the demonstrations to be directly applicable to downstream imitation learning policies. From the hardware perspective, we develop an updated exoskeleton built upon AirExo (Fang et al., 2024b), tailored for easy demonstration adaptation process. It provides a stronger mechanical structure and more accurate calibration, making the action capturing more precise and the system more robust. Comprehensive analyses are conducted to evaluate the efficiency and accuracy of the ***AirExo*-2** system in collecting demonstrations.

Beyond scaling up in-the-wild demonstration collection, developing a robust policy is crucial for effectively utilizing these demonstrations. To this end, we introduce ***RISE*-2**, a generalizable policy

that seamlessly integrates both 2D and 3D perceptions, as shown at the bottom of Fig. 1. Experiments show that ***RISE*-2** not only outperforms previous imitation learning policies in in-domain evaluations, but also surpasses prior generalizable policies in several out-of-domain scenarios, even when trained on a limited number of demonstrations from a narrow domain. Leveraging the strong generalization capabilities of ***RISE*-2**, we show that using in-the-wild demonstrations collected and adapted by the ***AirExo*-2** system, *without requiring additional demonstrations from the robot domain*, our policy achieves results comparable to, or even exceeding, those of imitation policies trained on an equivalent amount of teleoperated data.

2 Related Works

2.1 Scaling up Demonstration Collection

Demonstration data is essential for advancing imitation learning in robotic manipulation, as it serves as a foundation for learning complex and structured behaviors from expert demonstrations. Recent research on data scaling laws (Lin et al., 2024a) has revealed that similar scaling patterns emerge in imitation learning for robotic manipulation, analogous to those previously identified in natural language processing (Kaplan et al., 2020) and computer vision (Henighan et al., 2020; Peebles & Xie, 2023) fields. This highlights the importance of scaling up the demonstration collection.

Currently, there are four main directions for acquiring demonstration: teleoperation, generation in simulation, human video, and in-the-wild data collection.

Teleoperation. A straightforward method for collecting real-world robot demonstrations is human teleoperation, which directly captures demonstrations in the robot domain and is widely regarded as one of the most effective data collection techniques in real-world robotic imitation learning. However, it presents several challenges, especially when it comes to scalability. Scaling up teleoperated-based demonstration collection requires increasing the number of both robots and teleoperation devices, with robots being particularly costly to scale. Therefore, current large-scale robotic manipulation datasets collected through teleoperation (Brohan et al., 2023; Collaboration et al., 2024; Fang et al., 2024a; Jang et al., 2021; Khazatsky et al., 2024; Walke et al., 2023; Wu et al., 2024b) usually require significant human and physical resources in the data collection process. Another drawback is inefficiency, as teleoperating robots to complete tasks is far less intuitive than performing the tasks directly with human hands, leading to high learning costs (Luo et al., 2024) and suboptimal demonstrations (Chen et al., 2024a).

Generation in Simulation. This line of research addresses the demonstration scaling problem by automatically generating or augmenting demonstrations in simulation. Several studies generate demonstrations with large language models and skill-level agents (Hua et al., 2024; Mu et al., 2024; Wang et al., 2024f;g), while other approaches augment few human teleoperated demonstrations through replay (Ameperosa et al., 2024; Hoque et al., 2024; Jiang et al., 2024; Mandlekar et al., 2023; Wang et al., 2024d). Although these methods simplify the demonstration generation process, policies trained on such demonstrations often require additional sim-to-real adaptation before they can be applied in real-world scenarios.

Human Video. Researchers have also explored leveraging internet-scale human videos for robotic manipulation policy learning. Some approaches (Bahl et al., 2023; Ma et al., 2023a;b; Majumdar et al., 2023; Nair et al., 2022; Radosavovic et al., 2022; Srirama et al., 2024; Zeng et al., 2024) focus on visual representation learning from human videos, while others (Bharadhwaj et al., 2024a; Cheang et al., 2024; He et al., 2024; Qin et al., 2022; Wu et al., 2024a; Ye et al., 2024) propose pre-training the policy backbone with auxiliary video or latent prediction objectives. Since accurately extracting human hand states and 3D spatial trajectories from 2D videos remains challenging (McCarthy et al., 2024), it is still difficult to convert human videos into usable demonstrations for direct training the policies without fine-tuning on additional in-domain robot demonstrations.

In-the-Wild Data Collection. In-the-wild data refers to demonstrations collected by humans using specialized hardware devices, such as hand-held grippers (Chi et al., 2024; Etukuru et al., 2024; Seo et al., 2024; Shafiullah et al., 2023; Young et al., 2020), hand-held cameras (Duan et al., 2023; Wang et al., 2024c), VR/AR glasses (Chen et al., 2024c; Kareer et al., 2024; Kim et al., 2023a), motion-capture gloves (Wang et al., 2024a), and exoskeletons (Fang et al., 2024b; Kim et al., 2023a). These

devices act as a bridge between humans and robots, translating human hand motions into corresponding robot end-effector actions during demonstration collection. Without the dependency of physical robots, it is cost-effective for collecting in-the-wild demonstrations at scale. Nonetheless, challenges remain in improving the accuracy of such motion translation and addressing visual inconsistencies between humans and robots.

2.2 Learning from In-the-Wild Demonstrations

Despite promising in terms of scalability, two domain gaps pose obstacles in learning from in-the-wild demonstrations: the kinematic gap and the visual gap (Fang et al., 2024b; Kim et al., 2023a). The kinematic gap refers to the discrepancy in motion translation between humans and robots, where inaccuracies can affect action quality to some extent. The visual gap, on the other hand, pertains to the fact that visual information captured in in-the-wild demonstrations often includes specialized devices and human hands, whereas the visual information in robot demonstrations and deployments should contain the robot itself.

Kinematic Gap. The kinematic gap can be bridged using either visual or mechanical methods. DemoAT (Young et al., 2020) employs structure-from-motion (Schonberger & Frahm, 2016) to approximate the end-effector pose from a sequence of RGB images. Other visual methods leverage off-the-shelf pose estimation frameworks from commercial cameras, such as GoPro (Chi et al., 2024), iPhone Pro (Duan et al., 2023; Etukuru et al., 2024; Shafiullah et al., 2023; Wang et al., 2024c), RealSense T265 (Chen et al., 2024c; Seo et al., 2024; Wang et al., 2024a), and Aria glasses (Kareer et al., 2024). Mechanical methods typically build isomorphic devices (Fang et al., 2024b; Kim et al., 2023a) that obtain the robot poses from angle encoder readings.

Visual Gap. Hand-held devices (Chi et al., 2024; Etukuru et al., 2024; Seo et al., 2024; Shafiullah et al., 2023; Young et al., 2020) rely solely on in-hand cameras for visual observation and employ the same end-effector during robot deployment to prevent visual inconsistencies. Most other methods address the visual gap by incorporating additional real robot demonstration data for fine-tuning or co-training (Duan et al., 2023; Fang et al., 2024b; Kareer et al., 2024; Wang et al., 2024c), or by using human-in-the-loop techniques to collect corrective behaviors during policy deployment (Chen et al., 2024c; Wang et al., 2024a). M2R (Kim et al., 2023a) utilizes cropped observations with limited fields of view to reduce the impact of visual inconsistencies.

2.3 Generalizable Manipulation Policy

Direct learning from in-the-wild demonstrations emphasizes the need for a generalizable manipulation policy. Such a policy must effectively transfer the skills learned from in-the-wild demonstrations to the robot during real-world deployment. A generalizable policy is defined by its ability to adapt to new domains or environments, even when trained with limited demonstrations from a restricted domain (Xia et al., 2024). Specifically, it should be capable of generalizing across variations such as different camera perspectives, backgrounds, objects, and even embodiments.

Although many behavior-cloning-based (Pomerleau, 1988) robotic manipulation policies (Chi et al., 2023; Gervet et al., 2023; Goyal et al., 2023; Shridhar et al., 2022; Zhao et al., 2023) have demonstrated strong performance during in-domain evaluations, they often struggle in out-of-distribution scenarios, leading to compounded errors and task failures (Wang et al., 2024b; Xia et al., 2024). While large-scale pre-training on real-world robot demonstration data can improve the generalization ability of a robotic manipulation policy to some extent (Bharadhwaj et al., 2024b; Brohan et al., 2023; Collaboration et al., 2024; Jang et al., 2021; Kim et al., 2024; Liu et al., 2024a; Octo Model Team et al., 2024; Wang et al., 2024e; Zitkovich et al., 2023), it does not overcome the inherent upper bound of the policy's generalization ability, *i.e.*, its fundamental capacity to adapt across diverse domains and contexts in manipulation tasks.

Leveraging 3D perceptions (Chen et al., 2023b; Gervet et al., 2023; Goyal et al., 2023; Wang et al., 2024b; Ze et al., 2024) and 2D foundation models (Burns et al., 2023; Lin et al., 2024b; Qian et al., 2024; Xia et al., 2024; Zhang et al., 2024a) are two promising avenues towards generalizable manipulation policies. The former utilizes geometric cues to supplement the policy's understanding of the physical environment, while the latter harnesses the rich semantic features of 2D foundation models (Chen et al., 2021; Kirillov et al., 2023; Oquab et al., 2024; Radford et al., 2021; Rombach et al.,

2022) to improve the policy's ability to recognize and interpret complex object and scene information. Recent studies (Jia et al., 2024; Zhang et al., 2023b; 2024b) have explored combining these two approaches to further enhance policy performance. However, challenges remain in effectively integrating these insights for better generalization across diverse manipulation domains.

3 *AirExo-2*: Collecting and Adapting In-the-Wild Demonstrations

3.1 Overview

AirExo-2 prioritizes in-the-wild demonstration collection and subsequent data adaptation. Our goal is to *efficiently collect and adapt in-the-wild demonstrations into pseudo-robot demonstration for direct use in training real-world robotic manipulation policies*, which is particularly suitable for scaling up demonstration collection at a low cost. From this perspective, the main issues that need to be solved are:

- **D1**. The operation space from the in-the-wild demonstrations should be aligned with those of the robot, bridging the kinematic gap.
- **D2**. The visual observations from both in-the-wild demonstrations and robot deployment should be transformed into a unified domain, addressing the visual gap.

These two issues require a robust hardware and accurate calibration process to ensure the action capturing aligns well with the robot space, therefore facilitating subsequent demonstration adaptation. We detail the updated exoskeleton hardware design and calibration process in Appendix A.1 and Appendix A.2, respectively. The whole system retains the low-cost advantage of AirExo (Fang et al., 2024b), with the dual-arm demonstration collection platform (excluding the camera) priced at only $600. All hardware models, data collection code, and installation guides will be open-sourced. The demonstration adaptation process is introduced as follows.

3.2 Demonstration Adaptor

As discussed in §2.2, in-the-wild demonstrations often exhibit significant visual and kinematic differences from robot demonstrations, hindering their direct use for robotic manipulation policy learning. Prior approaches typically *avoid* the visual gap by using in-hand cameras or relying on pre-training and fine-tuning. In contrast, our ***AirExo-2*** system *solves* the visual gap by employing adaptors to convert in-the-wild observations into pseudo-robot observations. Together with an operation space adaptor that unifies the coordinate system of both demonstrations into the camera coordinates, these components form the demonstration adaptor. The demonstration adaptor helps align the in-the-wild data more closely with the robot's operating conditions, enabling the learned policies to be more *directly transferable* to real-world robotic manipulation tasks. Fig. 2 shows an overview of our demonstration adaptor.

Operation Space Adaptor (D1). Theoretically, we can transform the end-effector poses of both arms into the device base for both ***AirExo-2*** and the dual-arm robot platform, as they are morphologically identical. However, achieving such precision during the installation of the robotic arms is challenging, which means that we need to treat the dual-arm robot system as two separate single-arm robot systems in practice. This results in the device base not being a universal coordinate frame. Therefore, we opt to project all states and actions into the global camera coordinate system using the calibration results.

Image Adaptor (D2). With the recorded ***AirExo-2*** encoder readings and the calibrated transformation between the global camera and the ***AirExo-2*** base, we integrate the ***AirExo-2*** model into the Open3D rendering engine (Zhou et al., 2018). Using similar methods as previously described in the calibration section, we can render RGB-D and mask images of ***AirExo-2***. Due to the one-to-one joint mapping between ***AirExo-2*** and the dual-arm robot, we can also render the corresponding RGB-D and mask images of the dual-arm robot.

In addition to obtaining the ***AirExo-2*** mask from the renderer, we also need to address the visual information related to the human hands in the in-the-wild demonstrations. To achieve this, we use

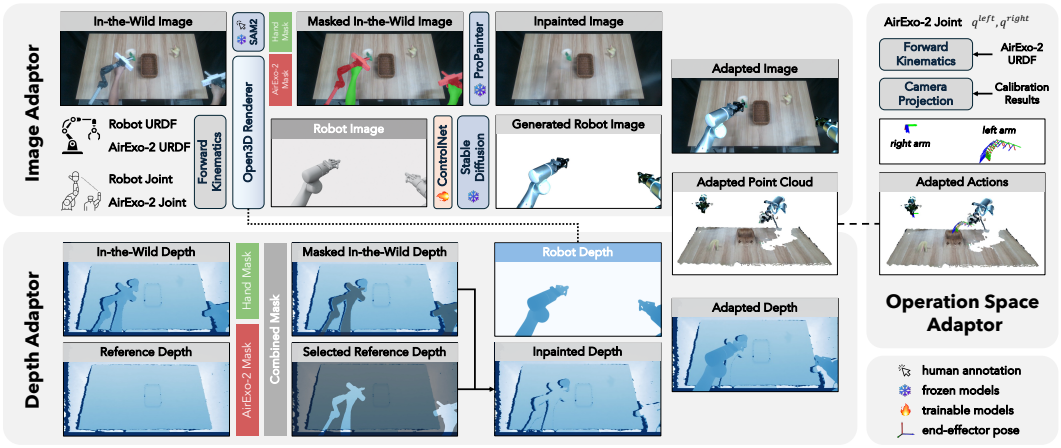


Figure 2: Illustration of the Demonstration Adaptor. We propose a demonstration adaptor to convert in-the-wild demonstrations into pseudo-robot demonstrations. It comprises three modules: an **operation space adaptor** for kinematic transformation, an **image adaptor** for visual processing, and a **depth adaptor** for depth adaptation.

SAM-2 (Ravi et al., 2024) to generate a consistent hand mask throughout the demonstration video. By merging this mask with the ***AirExo*-2** mask, we can identify the regions where the in-the-wild demonstration visually differs from the robot demonstration. Next, we apply the pre-trained video inpainting model ProPainter (Zhou et al., 2023) to fill in the masked areas, effectively removing the human embodiment information from the images to generate agent-agnostic images. Inspired by (Chen et al., 2024b), we then fine-tune a pre-trained Stable Diffusion 1.5 (Rombach et al., 2022) model with ControlNet (Zhang et al., 2023a) to generate photorealistic robot images from the rendered robot images. These generated robot images are then extracted using the robot mask and superimposed onto the inpainted images, producing the final pseudo-robot images.

Depth Adaptor (D2). For depth adaptation, we first capture a reference depth image of the empty workspace using the same camera setup, serving as a universal background reference. For each task, we identify static objects in the scene and record their depth values in the first frame as a demonstration-specific background reference. Using the merged mask provided by the image adaptor, we determine the regions of the depth map requiring adaptation and replace them with corresponding values from the demonstration-specific background. This process effectively removes depth information associated with human embodiment while preserving the scene's spatial consistency. Finally, by integrating the inpainted depth with the rendered robot depth, we can obtain the adapted depth.

4 *RISE*-2: A Generalizable Policy for Learning from In-the-Wild Demonstrations

4.1 Overview

Although we have transformed in-the-wild demonstrations into pseudo-robot demonstrations, inherent domain gaps such as differences in camera perspectives remain. Therefore, a generalizable policy is crucial for efficiently and effectively learning from these transformed in-the-wild demonstrations produced by the ***AirExo*-2** system.

As discussed in §2.3, 3D perception and 2D foundation models play complementary roles in creating a generalizable policy. 3D perception captures view-invariant geometric features of the scene, while 2D foundation models utilize their extensive knowledge to extract rich semantic features. Notably, 3D perception is especially beneficial for learning from in-the-wild demonstrations, as it explicitly infers spatial positions using a single camera, unlike 2D policies that often rely on multiple cameras to determine positions indirectly.

Building on these insights, we propose a 3D generalizable policy, ***RISE*-2**, to facilitate efficient learning from in-the-wild demonstrations and achieve robust task performance. Built upon

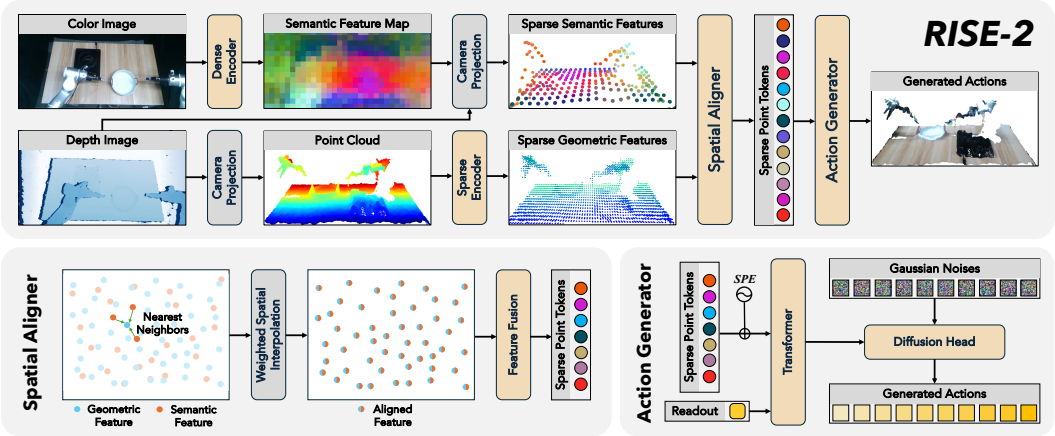


Figure 3: *RISE*-2 Policy Architecture. ***RISE*-2** takes an RGB-D observation as input and generates continuous actions in the camera frame. It is composed of four modules: 1) the color image is fed into the **dense encoder** to obtain semantic features organized in 2D form, which is then projected to sparse 3D form using reference coordinates; 2) the depth image is transformed to a point cloud and fed into the **sparse encoder** to obtain the local geometric features of seed points; 3) in the **spatial aligner**, the semantic features and the geometric features are aligned and fused using their 3D coordinates; 4) in the **action generator**, the fused features are converted to sparse point tokens, mapped to action space using a transformer and sparse positional encoding (*SPE*), and decoded into continuous actions by a diffusion head.

RISE (Wang et al., 2024b), our ***RISE*-2** policy addresses the following limitations of the original approach:

- **P1**. The raw point and color information are jointly encoded in RISE, causing geometric and semantic features to interfere with each other, which leads to consistent positional offsets in output actions when the background changes.
- **P2**. The inevitable noise in depth sensors often results in low-quality point cloud textures, making it challenging to extract rich semantic features solely from point cloud data and limiting the model's scene understanding capabilities.
- **P3**. The sparse encoder lacks pretraining on large-scale 3D scene datasets, hindering its ability to generalize across varying instances, backgrounds, and embodiments. Moreover, large-scale pretraining for 3D data introduces significant computational overhead, which is impractical for a shallow encoder.

Based on the above limitations, the design of ***RISE*-2** focuses on the precise feature fusion of 2D images and 3D point clouds to leverage the advantages of 2D vision in semantic information and 3D vision in spatial information simultaneously.

4.2 Policy Architecture

As shown in Fig. 3, ***RISE*-2** consists of four modules: a **sparse encoder** for 3D geometric feature extraction, a **dense encoder** for 2D semantic feature extraction, a **spatial aligner** for 2D-3D feature fusion and an **action generator** to decode visual features into actions. The implementation details are listed in Appendix A.5.

Sparse Encoder (P1). 3D point cloud data contains rich spatial structure information, which greatly facilitates the extraction of local geometric features. Such property has been successfully applied in general grasping (Fang et al., 2023; 2020; Wang et al., 2021). RISE featurizes the point cloud data with raw color information to obtain the semantic cues and geometric cues simultaneously, but fails to distinguish the coordinate shift and the color shift. ***RISE*-2** inherits the sparse 3D encoder (Choy et al., 2019) from RISE, but removes the color information to obtain pure geometric features. We denote this module by E_s, which implements the transformation from depth image to sparse geometric features:

$$\mathrm{E}_s : (\mathbf{D}, K) \rightarrow (\mathbf{F}_g, \mathbf{C}_g), \tag{1}$$

where $\mathbf{D}$ denotes the observed depth image, K denotes the corresponding camera intrinsic, and $\mathbf{C}_g = \{c_i^g \in \mathbf{P}\}$ denotes the seed points after network down-sampling. $\mathbf{D}$ is firstly converted to a point cloud $\mathbf{P}$ using camera intrinsic K. The sparse network takes $\mathbf{P}$ as input and extracts the local geometric features $\mathbf{F}_g = \{f_i^g\}$, where f_i^g is the corresponding feature vector of c_i^g. This lightweight encoder enhances the efficiency of sparse feature extraction, ensuring the real-time performance of the policy.

Dense Encoder (P2 and P3). A generalizable policy requires rich semantic features to understand the scene, while the low-quality texture of point cloud data poses a challenge to this demand. Unlike RISE, ***RISE*-2** adopts a dense 2D encoder to process organized color information, which is denoted by E_d. E_d implements the transformation from color image to dense semantic features:

$$\mathrm{E}_d : (\mathbf{I}, \mathbf{D}, K) \rightarrow (\mathbf{F}_s, \mathbf{C}_s), \tag{2}$$

where $\mathbf{I}$ denotes the observed color image and $\mathbf{F}_s = \{f_i^s\}$ denotes the output semantic feature map with the width w and the height h. Since $\mathbf{F_s}$ is densely organized in 2D form, we also compute its reference 3D coordinates $\mathbf{C}_s = \{c_i^s\}$ for the mapping to sparse form. Let the raw point cloud $\mathbf{P}$ be organized in 2D form, $\mathbf{C}_s$ is computed by:

$$\mathbf{C}_s = \mathrm{AdaptiveAvgPool2d}(\mathbf{P}, [w, h]), \tag{3}$$

where $\mathrm{AdaptiveAvgPool2d}(\cdot, [w, h])$ applies an average pooling function to $\mathbf{P}$, and the output shape is $w \times h$. By leveraging continuous color information distributed in high-resolution data, E_d significantly enhances the policy's ability to capture the details in the scene.

One significant advantage of the dense encoder is the usage of visual foundation models, offering highly generalized visual representations that excel across diverse tasks and domains (Xia et al., 2024). ***RISE*-2** employs DINOv2 (Oquab et al., 2024) fine-tuned with LoRA (Hu et al., 2022) to implement E_d. Such design improves the policy's robustness and adaptability in understanding contextual relationships within the environments.

Spatial Aligner (P1). ***RISE*-2** extracts the geometric features and the semantic features using separate encoders, posing a challenge for the fusion of features distributed in different domains. One solution is to directly concatenate the two aggregated feature vectors, but it loses fine-grained local features which are vital for precise perception. Another alternative upsamples $\mathbf{F}_s$ to the size of the original dense image $\mathbf{I}$, projects it to the point cloud $\mathbf{P}$, and downsamples the features to align with the seed points $\mathbf{C}_g$ (Zhang et al., 2023b). This approach incurs a high computational cost, decreasing the efficiency of policy training and deployment.

Instead, ***RISE*-2** utilizes a spatial aligner to efficiently fuse the two kinds of features based on their 3D coordinates $\mathbf{C}_g$ and $\mathbf{C}_s$. For a point $c_i^g \in \mathbf{C}_g$ output by the sparse encoder E_s, we compute its nearest neighbors $\mathbf{N}_i = \{n_j^i | j = 1, \cdots, M\}$ from $\mathbf{C}_s$ output by the dense encoder E_d. The semantic feature of c_i^g is computed by weighted spatial interpolation:

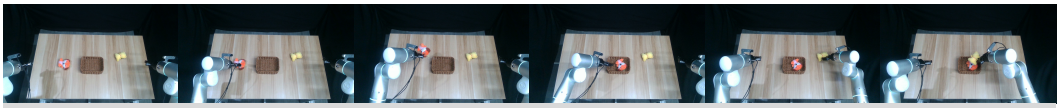
$$f_i^* = \frac{\Sigma_{j=1}^M f_j^s / \mathrm{dist}(c_i^g, n_j^i)}{\Sigma_{j=1}^M 1 / \mathrm{dist}(c_i^g, n_j^i)}, \tag{4}$$

where $\mathrm{dist}(c_i, c_j)$ is the Euclidean distance between c_i and c_j. The aligned feature of point c_i^s is

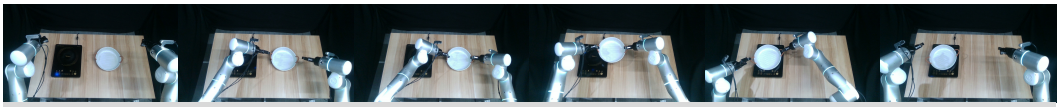
$$f_i = \mathrm{Concat}(f_i^g, f_i^*). \tag{5}$$

By aligning the seed points $\mathbf{C}_g$ with the dense feature map $\mathbf{F}_s$ using 3D coordinates, we can easily obtain the precise semantic features of points in different locations. The aligned features are then fused to high-level sparse representations using sparse convolution layers (Choy et al., 2019). The visualization results of applying weighted spatial interpolation to the 2D feature map can be found in Appendix A.8.

Action Generator. The action generator adopts a similar architecture to RISE, which uses a transformer (Vaswani et al., 2017) to approximate the mapping from point features with sparse positional encoding to the action space, and a diffusion head (Chi et al., 2023; Ho et al., 2020b; Janner et al., 2022) to generate the action chunk (Zhao et al., 2023). The transformer in ***RISE*-2** is in a decoder-only form, taking sparse point tokens and a readout token as input. Conditioning on the feature of the readout token, the diffusion head decodes the Gaussian noises into continuous actions. The generated actions are in the camera frame to ensure consistency across different scenes and camera views. The translations are in absolute positions and the rotations are in 6D representation (Zhou et al., 2019).



Collect Toys. Two cotton toys are placed on the left and right sides of a table, with a square basket in the center. The goal is to pick up each toy with the corresponding arm and place it into the basket.



Lift Plate. An induction cooker and a double-handle plate are placed on the table. The goal is to use both arms to grasp the handles of the plate, lift it, and place it onto the cooker.

Figure 4: Tasks. We design two tasks to evaluate the in-domain and generalization capabilities of the ***RISE*-2** policy. Additionally, we assess the ability of the ***AirExo*-2** system to transform high-quality pseudo-robot demonstrations derived from in-the-wild data. These transformed demonstrations are then used to train downstream policies, allowing us to evaluate their transferability to real robot platforms.

5 EXPERIMENTS

In this section, we aim to answer the following research problems.

- **Q1**. Are in-hand cameras sufficient for effectively perceiving and executing manipulation tasks?
- **Q2**. Does ***RISE*-2** outperform previous policies in in-domain evaluations?
- **Q3**. Can ***RISE*-2** generalize to environmental disturbances, such as unseen objects and backgrounds?
- **Q4**. Can generalizable policies like ***RISE*-2**, trained exclusively on pseudo-robot demonstrations collected and transformed by the ***AirExo*-2** system, be directly deployed on a real robot?
- **Q5**. How important is the demonstration adaptor of the ***AirExo*-2** system for transferring policies trained solely on in-the-wild demonstrations to a real robot platform?

5.1 SETUP

Platform. Our dual-arm robot platform uses two Flexiv Rizon 4 robotic arms, each equipped with a Robotiq 2F-85 gripper. An Intel RealSense D415 camera is mounted on top of the robot platform to capture global observations, while two additional Intel RealSense D415 cameras are mounted on the wrists of each robot arm to provide in-hand observations for 2D image-based policies as additional views.

Tasks. As shown in Fig. 4, we designed two tasks to evaluate the in-domain and generalization performance of the ***RISE*-2** policy, as well as to assess the overall effectiveness of the ***AirExo*-2** system in transferring policies of different tasks trained on in-the-wild demonstrations to a real robot.

Data Collection. The teleoperated demonstrations are collected using AirExo (Fang et al., 2024b), while the in-the-wild demonstrations are gathered and transformed through our proposed ***AirExo*-2** system. For each task, we collect 50 teleoperated demonstrations for policy evaluation and 50 in-the-wild demonstrations to test whether a generalizable policy can be zero-shot deployed to the robot platform using the in-the-wild demonstrations collected and processed by ***AirExo*-2**.

Baselines. We compare ***RISE*-2** against a range of representative policies based on 2D images and 3D point clouds, including: (1) **ACT** (Zhao et al., 2023), which employs transformers to map image observations and proprioception to robot action chunks; (2) **Diffusion Policy (DP)** (Chi et al., 2023), which formulates action prediction as a diffusion denoising process (Ho et al., 2020a; Song et al., 2021) conditioned on the image observations; (3) **CAGE** (Xia et al., 2024), an extension of DP that incorporates visual foundation models (Oquab et al., 2024), a causal observation perceiver (Jaegle et al., 2022), and an attention-based diffusion action head for improved generalization; and (4) **RISE** (Wang et al., 2024b), a 3D imitation policy that leverages a sparse 3D encoder for efficient point cloud perception.

Evaluation Protocols. All policies are deployed on a workstation with an NVIDIA RTX 2060 SUPER GPU. Following the procedure outlined in (Chi et al., 2024; Xia et al., 2024), we adopt a consistent evaluation method for each policy to minimize performance variation and ensure reproducibility. Specifically, we generate uniformly distributed test positions randomly before each task evaluation. The workspace is set up identically across different policies and test environments, and success rates are recorded for each test case. Each policy is evaluated over 20 consecutive trials per task, and the success rates are computed accordingly.

5.2 Case Study: Are In-Hand Cameras Sufficient?

We conducted a case study to investigate whether in-hand cameras are sufficient for many manipulation tasks. We select the ***Collect Toys*** task as an example and utilize CAGE (Xia et al., 2024) as the policy for this case study.

Method	# Cameras	Success Rate
CAGE (global only)	1	45.0%
CAGE (in-hand only)	2	60.0%
CAGE (global + in-hand)	3	72.5%

Table 1: Case Study Results. In this case study, we employ CAGE with relative action representations following its original implementation Xia et al. (2024). The in-domain and generalization experiments afterward will use CAGE with absolute action representations. For details about action representations, please refer to Appendix A.6.

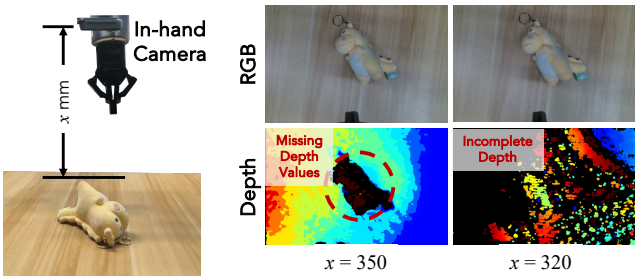


Figure 5: Visualization of In-hand Camera Observation. In-hand cameras often produce low-quality depth observations during interactions with objects, limiting their usages for most 3D point cloud-based policies.

In-hand cameras alone are often insufficient for manipulation tasks and may pose additional obstacles on policy learning (Q1). As shown in Tab. 1, neither global nor in-hand cameras alone provide adequate observations for 2D image-based policies to achieve strong performance. Recent work (Wang et al., 2024b) has demonstrated that using only a global camera enables a 3D imitation policy to outperform 2D multi-view image-based policies, highlighting the importance of 3D information for scene understanding. However, as illustrated in Fig. 5, in-hand cameras may produce incomplete depth information when the robotic arm approaches an object, making them unsuitable for 3D point-cloud-based policies. Consequently, relying solely on in-hand cameras can degrade the performance of the policies, particularly for 3D policies that rely on complete and accurate depth information to achieve superior performance.

5.3 Policy In-Domain Evaluation: ***RISE*-2**

RISE*-2 achieves significantly better performance than previous policies during in-domain evaluations (Q2).** Tab. 2 reports the success rates for both tasks, highlighting the effectiveness of ***RISE*-2** in handling diverse manipulation challenges. In the ***Collect Toys task, ***RISE*-2** consistently outperforms all baselines across both arms, leading to a substantially higher overall success rate. This shows the capability of RISE in achieving Similarly, in the ***Lift Plate*** task, which requires precise motion execution, ***RISE*-2** demonstrates superior accuracy in predicting fine-grained robotic actions, surpassing all baselines. These results indicate that ***RISE*-2** not only improves overall task success but also enhances control precision, making it well-suited for complex manipulation scenarios.

Method	*Collect Toys*			*Lift Plate*	
	Overall	Left	Right	Grasp	Place
ACT (Zhao et al., 2023)	32.5%	50%	15%	45%	20%
DP (Chi et al., 2023)	40.0%	25%	55%	30%	30%
CAGE (Xia et al., 2024)	65.0%	70%	60%	55%	55%
RISE (Wang et al., 2024b)	72.5%	60%	85%	75%	75%
***RISE*-2** *(ours)*	**95.0%**	**90%**	**100%**	**85%**	**85%**

Table 2: In-Domain Evaluation Results of Different Policies on the *Collect Toys* and the *Lift Plate* Task. Our ***RISE*-2** policy outperforms baselines during in-domain evaluations.

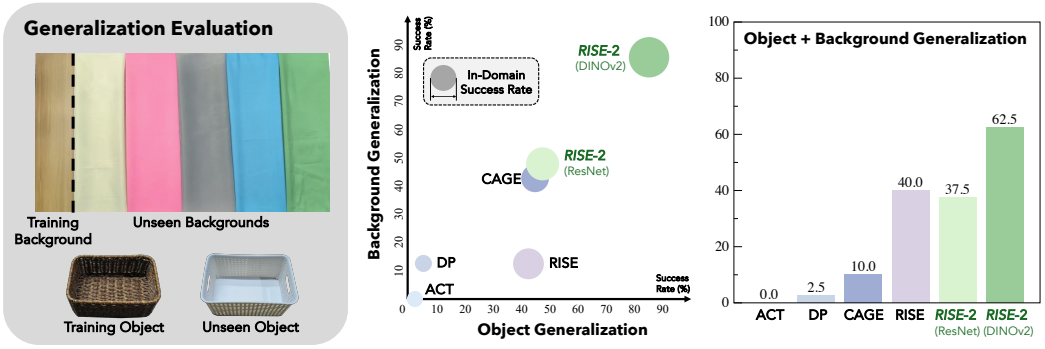


Figure 6: Generalization Evaluation Setup and Results. (Left) We use five unseen backgrounds and one unseen target object to evaluate the generalization ability of the imitation policies. The policies are trained on demonstrations in a narrow domain, which consists of a single training background and target object. (Middle) When only the object or background is replaced, ***RISE*-2** maintains strong performance with a minimal drop of 10%. (Right) In more challenging scenarios, where both the object and background are replaced with unseen ones, ***RISE*-2** still demonstrates notable generalization capabilities.

The improvement of ***RISE*-2** over RISE also showcases the significance of using separate 2D and 3D encoders along with spatial feature alignment. This design effectively decouples geometric and semantic feature extraction, allowing them to be seamlessly integrated through coordinate-based fusion via spatial aligner, leading to more accurate and robust representations for manipulation tasks.

5.4 Policy Generalization Evaluation: ***RISE*-2**

We select the ***Collect Toys*** task to conduct a generalization experiment to evaluate the robustness of different policies under varying levels of environmental disturbances. As illustrated in Fig. 6 (left), we introduce two types of disturbances: background variations and object differences. To further investigate the role of disentangling geometric and semantic features in generalization and assess the impact of different visual backbones for semantic feature extraction, we include a variant of ***RISE*-2** that replaces the DINOv2 encoder with a ResNet-18 encoder (He et al., 2016). We then follow the same evaluation protocol to compute success rates, enabling a comprehensive comparison of each policy's generalization capability.

***RISE*-2 exhibits strong generalization performance to different environmental disturbances (Q3).** As shown in Fig. 6 (middle), under single disturbances such as background or object replacement, ***RISE*-2** maintains high performance, experiencing only a 10% success rate drop while still significantly outperforming previous methods. Notably, even when the DINOv2 encoder is replaced with a vanilla ResNet-18 encoder, although the performance is lower than the original ***RISE*-2**, it still demonstrates a reasonable level of generalization ability and surpasses baseline policies. This result further validates our design choice of employing separate encoders for 3D geometric and 2D semantic feature extraction, effectively enhancing policy robustness. To achieve even better generalization performance under disturbances, using visual foundation models like DINOv2 (Oquab et al., 2024) as 2D dense encoders is essential, as they can leverage the extensive semantic knowledge acquired through large-scale pre-training. This allows for the extraction of more generalizable features from the manipulation scene, ultimately elevating the generalization ability of ***RISE*-2**.

***RISE*-2 even shows decent generalization performance when facing a combination of disturbances (Q3).** The results in Fig. 6 (right) show that ***RISE*-2** retains its generalization performance to a large extent even under combined disturbances. RISE (Wang et al., 2024b) performs unexpectedly well under the combined disturbances, even surpassing its performance when only object replacement is applied. By observing the experimental process, we hypothesize that different disturbances may introduce different offsets in the predicted action of RISE, and sometimes, combining these disturbances may cause the offsets to cancel each other out, leading to unexpectedly good performance. Instead, ***RISE*-2** does not exhibit this phenomenon, as it consistently demonstrates strong performance across all types of generalization experiments, regardless of the disturbance combination.

5.5 System Evaluation: *AirExo*-2

Our previous experiments have shown that ***RISE*-2** is a generalizable policy, making it ideal for learning from in-the-wild demonstrations collected and transformed by ***AirExo*-2**. Accordingly, we train the policy using the pseudo-robot demonstrations processed by ***AirExo*-2**, and then zero-shot deploy the trained policy on the dual-arm robot platform, without using any additional robot demonstrations. For comparison, we also include RISE (Wang et al., 2024b) in this experiment.

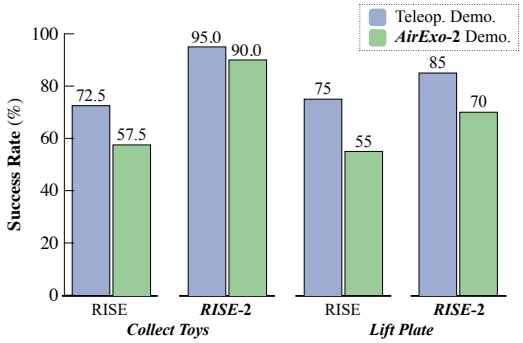


Figure 7: System Evaluation Results. Trained using demonstrations collected and adapted by the ***AirExo*-2** system, without any access to robot data, the policies maintain reasonable performance, highlighting the overall effectiveness of the ***AirExo*-2** system.

The *AirExo*-2 system provides high-quality pseudo-robot demonstrations that can be directly used to train generalizable policies like *RISE*-2, enabling successful zero-shot deployment of the trained policies to the real robot platform with reasonable performance (Q4). As shown in Fig. 7, ***RISE*-2** performs well when trained solely on pseudo-robot demonstrations collected and transformed by the ***AirExo*-2** system. The policy achieves satisfactory success rates for both tasks, with only a slight performance drop compared to learning from teleoperated demonstrations. Another policy, RISE, also shows good performance when deployed zero-shot, though with a slightly larger performance drop than ***RISE*-2**. These results underscore the importance of having a generalizable policy for transferring manipulation skills learned from in-the-wild demonstrations to real robot environments, in the absence of teleoperation data.

Method	**Success Rate** (%)	
	w.o. adaptor	*w.* adaptor
RISE Wang et al. (2024b)	30.0%	57.5%
***RISE*-2** *(ours)*	52.5%	90.0%

Table 3: Ablation Results of Demonstration Adaptors. While generalizable policies trained with raw in-the-wild demonstrations can occasionally transfer to real robot platforms, the inherent embodiment gap hinders their performance. Our proposed demonstration adaptor, especially visual adaptors, effectively bridges this domain gap and enhances the performance of policies during direct transfer, demonstrating its importance in this process.

Figure 8: Qualitative Results of the *Serve Steak* Task. The system is evaluated on a challenging, long-horizon, and contact-rich task, ***Serve Steak***, in which the robot needs to scoop a steak from a pan using a spatula and slide it onto a plate. This task requires precise control across multiple complex steps. We first collect in-the-wild human demonstrations (top) and convert them into pseudo-robot demonstrations (middle) using ***AirExo*-2**. These demonstrations are then used to train the ***RISE*-2** policy, which is successfully deployed on a real robot to complete the task autonomously (bottom).

The demonstration adaptor of *AirExo*-2 are necessary for achieving satisfactory policy transfer (Q5). We use the ***Collect Toys*** task to illustrate the importance of the demonstration adaptor in learning from in-the-wild demonstrations. Operation space adaptors are necessary for policy transfer, so we include them in each variant and ablate whether to use visual adaptors (image adaptor and depth adaptor) to transform visual observations into the robot domain. As shown in Tab. 3, performance drops significantly when learning directly from raw in-the-wild demonstrations without visual adaptors, indicating that the visual gap between in-the-wild and robot demonstrations is substantial and cannot be ignored. Despite this gap, our ***RISE*-2** policy still achieves performance comparable to RISE even without demonstration adaptors (52.5% *v.s.* 57.5%), showcasing its strong generalization ability in learning from cross-embodiment data. These results also validate the authenticity and reliability of the pseudo-robot demonstrations transformed by ***AirExo*-2**, confirming that they accurately capture real-world interactions and are effective for training generalizable policies. This highlights the potential of combining ***RISE*-2** and ***AirExo*-2** as a scalable framework for imitation learning using in-the-wild demonstrations.

5.6 Qualitative Results on Challenging Tasks

We additionally evaluate the whole system on a challenging long-horizon and contact-rich task ***Serve Steak***, as shown in Fig. 8. This task involves multiple challenging steps that require intricate robot actions: (1) grasp the plate using the left arm; (2) grasp the spatula using the right arm; (3) scoop up the steak in a pan with the grasped spatula; (4) lift the steak with the spatula and slide it onto the plate.

The integration of the *AirExo*-2 system with the *RISE*-2 policy enables the robot to tackle the challenging, long-horizon, and contact-rich task without requiring robot demonstrations, highlighting its potential for a wide range of manipulation tasks (Q6). After training ***RISE*-2** with only 50 in-the-wild demonstrations collected and processed by ***AirExo*-2**, we successfully deployed the policy on a real-world robot platform, where it can complete the entire task automatically with no robot data, as illustrated in Fig. 8. This result underscores the effectiveness of the proposed system in learning from a limited number of human demonstrations while generalizing to real-world scenarios, demonstrating its applicability to various complex manipulation tasks.

Notably, we expect the system's performance to further improve as we scale up in-the-wild demonstrations. The synergy between scalable data collection (***AirExo*-2**) and generalizable policy learning (***RISE*-2**) suggests a promising trajectory toward more efficient and generalizable robotic manipulation. Expanding this approach could enable autonomous robots to acquire increasingly complex skills with minimal human supervision, unlocking new possibilities for deployment in real-world environments with high variability and task diversity.

6 System Analysis

In this section, we conduct thorough analyses of our proposed ***AirExo*-2** system, including:

- **A1**. Is ***AirExo*-2** easy to use for in-the-wild data collection?
- **A2**. How does the data collection speed of ***AirExo*-2** compare to the data collection speed of teleoperation?
- **A3**. How accurate is the ***AirExo*-2** system in recording actions compared to previous handheld devices used for in-the-wild demonstrations?

6.1 User Study

We conduct a user study to comprehensively evaluate the intuitiveness and data throughput of several demonstration collection methods, including end-effector pose teleoperation with haptic device (Fang et al., 2024a), joint-space teleoperation with AirExo (Fang et al., 2024b), and ***AirExo*-2** in-the-wild data collection. The study involves 20 participants with varying levels of experience in robot demonstration collection, including 14 men and 6 women, aged between 21 and 35 years old. The participants are asked to collect one demonstration for the ***Collect Toys*** task using all three data collection platforms mentioned above. Before collection, participants are given 3 minutes to familiarize themselves with each data collection platform. We then record the time each participant spends collecting one demonstration. After completing the collection, we designed a questionnaire to gather their feedback on the three data collection methods.

Method	**Completion Time** (s) ↓	**Average Rank** ↓		**Preference Score** ↑
		Intuitiveness	**Learnability**	
EE pose teleop	$46.06_{\pm 27.21}$	3.00 / 3	2.95 / 3	29.75
joint-space teleop	$17.31_{\pm 5.055}$	1.80 / 3	2.00 / 3	49.58
***AirExo*-2** (ours)	$\mathbf{5.66}_{\pm 1.978}$	**1.20** / 3	**1.05** / 3	**83.00**

Table 4: User Study Results. Collecting in-the-wild demonstrations with ***AirExo*-2** enhances intuitiveness and enables higher data throughput. Users assign higher preference scores to ***AirExo*-2** compared to other teleoperation data collection methods.

The *AirExo*-2 system is intuitive and user-friendly, making it a good choice for large-scale demonstration collection (A1). From Tab. 4 we can observe that both experienced participants and novices in demonstration collection find ***AirExo*-2** more intuitive and easier to learn than teleoperation. Participants significantly prefer ***AirExo*-2** for in-the-wild collection over both joint-space and end-effector pose teleoperation. This ease of use translates to faster onboarding and smoother operation, making it an excellent tool for diverse users and ensuring more efficient data collection in real-world environments. We believe that the accessibility of ***AirExo*-2** contributes to more consistent and high-quality demonstrations, reinforcing its value for large-scale, real-world tasks.

Collecting demonstrations with *AirExo*-2 is more efficient compared to teleoperation (A2). We assume that the task completion times for different data collection methods follow Gaussian distributions, which is supported by the Shapiro-Wilk test results. To compare the efficiency of in-the-wild demonstration collection with ***AirExo*-2** versus teleoperated demonstration collection with AirExo, we conduct Welch's t-test. Based on the results presented in Tab. 4, we find that ***AirExo*-2** significantly outperforms AirExo joint-space teleoperation in terms of time efficiency, with a p-value of $8.32 \times 10^{-10} < 0.001$. Additionally, end-effector pose teleoperation is found to be the least efficient for collecting demonstrations. This highlights the advantage of *AirExo*-2 in streamlining the data collection process and improving the overall efficiency of task demonstrations.

6.2 ACCURACY ANALYSIS

Action accuracy is crucial for directly learning from in-the-wild demonstrations, ensuring that the policy can learn precise actions necessary for successful task execution. Handheld devices (Chi et al., 2024; Etukuru et al., 2024; Seo et al., 2024; Shafiullah et al., 2023; Young et al., 2020) typically rely on visual SLAM for camera pose trajectory estimation, whereas our ***AirExo-2*** system leverages its mechanical design and forward kinematics to calculate the robot end-effector trajectory. In this analysis, we select UMI (Chi et al., 2024) as a representative handheld data collection device and compare its action accuracy with that of our ***AirExo-2*** system. We have designed 3 tracks to evaluate the translation accuracies of both systems. Please refer to Appendix A.7 for more details.

Device	**Average Error** (mm) ↓			**Max Error** (mm) ↓
	Track 1	Track 2	Track 3	
UMI (Chi et al., 2024)	$7.476_{\pm 1.840}$	$10.665_{\pm 4.543}$	$8.360_{\pm 2.381}$	20.002
AirExo-2 *(ours)*	$\mathbf{1.213}_{\pm 1.332}$	$\mathbf{1.952}_{\pm 0.744}$	$\mathbf{1.903}_{\pm 1.736}$	**6.134**

Table 5: Action Accuracies of Different In-the-Wild Demonstration Collection Systems. ***AirExo-2*** exhibits superior action accuracies compared to handheld devices like UMI.

The *AirExo*-2 system demonstrates superior accuracy in recording actions compared to handheld devices, making it well-suited for in-the-wild demonstration collection across a wide range of manipulation tasks (A4). The error results in Tab. 5 show that ***AirExo-2*** achieves significantly lower action errors (approximately 2mm on average) compared to UMI (Chi et al., 2024), which relies on visual SLAM and IMU sensors for camera pose estimation. This highlights the advantage of ***AirExo-2***, as its mechanical design and forward kinematics provide higher precision than vision-based SLAM methods. These findings confirm that ***AirExo-2*** is a reliable and accurate tool for capturing high-fidelity motion data, making it an effective solution for large-scale in-the-wild demonstration collection, particularly for fine-grained manipulation tasks that require high precision.

7 LIMITATIONS AND FUTURE WORKS

While we utilize the proposed demonstration adaptor to visually transform in-the-wild demonstrations collected by ***AirExo-2*** into pseudo-robot demonstrations, these transformed demonstrations are primarily useful for generalizable policies. To enhance the applicability of the pseudo-robot demonstrations to a broader range of policies, future work could explore the integration of demonstration augmentation methods, such as novel view synthesis (Chen et al., 2024b; Sargent et al., 2024; Tian et al., 2024; Van Hoorick et al., 2025), into the demonstration adaptor to improve the diversity of the demonstrations, making them more versatile for various policy learning.

As demonstrated by several works (Chi et al., 2024; Hsu et al., 2022; Kim et al., 2023b; Young et al., 2020), the in-hand image is a semi-unified observation modality across different embodiments. Our case study also reveals that combining in-hand observations can strengthen the performance of various 2D policies. However, the current ***AirExo-2*** system does not include in-hand cameras. Although we have designed connectors to integrate them with the exoskeleton, calibrating the in-hand cameras with the exoskeleton remains challenging, making it difficult to adapt the in-hand images into the robot domain. Future work could explore effective methods for adapting in-hand images collected by ***AirExo-2*** to the robot domain, or investigate strategies for leveraging the semi-unified in-hand observations in policy design.

Another limitation lies in the end-effector. The current ***AirExo-2*** system only supports parallel grippers as end-effectors, limiting its applicability in more dexterous tasks. Future work could integrate the ***AirExo-2*** system with dexterous hands and their corresponding exoskeletons, enabling more complex manipulation capabilities and expanding the range of tasks the system can effectively perform.

8 CONCLUSION

This paper introduces ***AirExo*-2**, a novel system designed for large-scale in-the-wild demonstration collection and adaptation using low-cost exoskeletons. By incorporating a demonstration adaptor, ***AirExo*-2** enables the visual and kinematic transformation of in-the-wild demonstrations into pseudo-robot demonstrations, which can then be directly applied to downstream imitation learning tasks. We also propose a generalizable policy, ***RISE*-2**, which effectively integrates both 2D and 3D perception, demonstrating exceptional performance in both in-domain and out-of-domain scenarios.

Further experiments demonstrate that when trained exclusively on pseudo-robot demonstrations generated by the ***AirExo*-2** system — without using any robot demonstrations — the policy achieves satisfactory performance during zero-shot deployment on a real-world robot platform. This highlights the potential of combining ***AirExo*-2** and ***RISE*-2** as a scalable and promising alternative to traditional teleoperation-imitation pipelines, providing a more efficient, cost-effective solution for large-scale, generalizable robotic imitation learning. Together, these results open new possibilities for transferring manipulation skills from in-the-wild environments to real robots, without the need for extensive robot-centric data collection.

REFERENCES

Ezra Ameperosa, Jeremy A Collins, Mrinal Jain, and Animesh Garg. Rocoda: Counterfactual data augmentation for data-efficient robot learning from demonstrations. *arXiv preprint arXiv:2411.16959*, 2024.

Shikhar Bahl, Russell Mendonca, Lili Chen, Unnat Jain, and Deepak Pathak. Affordances from human videos as a versatile representation for robotics. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Homanga Bharadhwaj. Position: scaling simulation is neither necessary nor sufficient for in-the-wild robot manipulation. In *Forty-first International Conference on Machine Learning*, 2024.

Homanga Bharadhwaj, Debidatta Dwibedi, Abhinav Gupta, Shubham Tulsiani, Carl Doersch, Ted Xiao, Dhruv Shah, Fei Xia, Dorsa Sadigh, and Sean Kirmani. Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation. *arXiv preprint arXiv:2409.16283*, 2024a.

Homanga Bharadhwaj, Jay Vakil, Mohit Sharma, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking. In *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*, pp. 4788–4795. IEEE, 2024b.

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.

Gary Bradski. The opencv library. *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, 25(11):120–123, 2000.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong T. Tran, Vincent Vanhoucke, Steve Vega, Quan Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-1: robotics transformer for real-world control at scale. In *Robotics: Science and Systems*, 2023.

Kaylee Burns, Zach Witzel, Jubayer Ibn Hamid, Tianhe Yu, Chelsea Finn, and Karol Hausman. What makes pre-trained visual representations successful for robust manipulation? *arXiv preprint arXiv:2312.12444*, 2023.

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, et al. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024.

Jingjing Chen, Hongjie Fang, Hao-Shu Fang, and Cewu Lu. Towards effective utilization of mixed-quality demonstrations in robotic manipulation via segment-level selection and optimization. *arXiv preprint arXiv:2409.19917*, 2024a.

Lawrence Yunliang Chen, Chenfeng Xu, Karthik Dharmarajan, Muhammad Zubair Irshad, Richard Cheng, Kurt Keutzer, Masayoshi Tomizuka, Quan Vuong, and Ken Goldberg. Rovi-aug: Robot and viewpoint augmentation for cross-embodiment robot learning. In *Conference on Robot Learning (CoRL)*, 2024b.

Linghao Chen, Yuzhe Qin, Xiaowei Zhou, and Hao Su. Easyhec: Accurate and automatic hand-eye calibration via differentiable rendering and space exploration. *IEEE Robotics and Automation Letters*, 2023a.

Shizhe Chen, Ricardo Garcia Pinel, Cordelia Schmid, and Ivan Laptev. Polarnet: 3d point clouds for language-guided robotic manipulation. In *Conference on Robot Learning*, pp. 1761–1781. PMLR, 2023b.

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*, 2024c.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pp. 9620–9629. IEEE, 2021.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023.

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.

Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3075–3084, 2019.

Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models. In *IEEE International Conference on Robotics and Automation*, pp. 6892–6903, 2024.

Jiafei Duan, Yi Ru Wang, Mohit Shridhar, Dieter Fox, and Ranjay Krishna. AR2-D2: training a robot without a robot. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2838–2848. PMLR, 2023.

Haritheja Etukuru, Norihito Naka, Zijin Hu, Seungjae Lee, Julian Mehu, Aaron Edsinger, Chris Paxton, Soumith Chintala, Lerrel Pinto, and Nur Muhammad Mahi Shafiullah. Robot utility models: General policies for zero-shot deployment in new environments. *arXiv preprint arXiv:2409.05865*, 2024.

Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11444–11453, 2020.

Hao-Shu Fang, Chenxi Wang, Hongjie Fang, Minghao Gou, Jirong Liu, Hengxu Yan, Wenhai Liu, Yichen Xie, and Cewu Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics*, 2023.

Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. RH20T: A comprehensive robotic dataset for learning diverse skills in one-shot. In *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*, pp. 653–660. IEEE, 2024a.

Hongjie Fang, Hao-Shu Fang, Yiming Wang, Jieji Ren, Jingjing Chen, Ruo Zhang, Weiming Wang, and Cewu Lu. Airexo: Low-cost exoskeletons for learning whole-arm manipulation in the wild. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 15031–15038. IEEE, 2024b.

Théophile Gervet, Zhou Xian, Nikolaos Gkanatsios, and Katerina Fragkiadaki. Act3d: 3d feature field transformers for multi-task robotic manipulation. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pp. 3949–3965. PMLR, 2023.

Ankit Goyal, Jie Xu, Yijie Guo, Valts Blukis, Yu-Wei Chao, and Dieter Fox. RVT: robotic view transformer for 3d object manipulation. In *Conference on Robot Learning*, pp. 694–710. PMLR, 2023.

Haoran He, Chenjia Bai, Ling Pan, Weinan Zhang, Bin Zhao, and Xuelong Li. Large-scale actionless video pre-training via discrete diffusion for efficient policy learning. *arXiv preprint arXiv:2402.14407*, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

Nick Heppert, Max Argus, Tim Welschehold, Thomas Brox, and Abhinav Valada. Ditto: Demonstration imitation by trajectory transformation. *arXiv preprint arXiv:2403.15203*, 2024.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020a.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020b.

Zhengdong Hong, Kangfu Zheng, and Linghao Chen. Fully automatic hand-eye calibration with pretrained image models. *International Conference on Intelligent Robots and Systems (IROS)*, 2024.

Ryan Hoque, Ajay Mandlekar, Caelan Garrett, Ken Goldberg, and Dieter Fox. Intervengen: Interventional data generation for robust and data-efficient robot imitation learning. *arXiv preprint arXiv:2405.01472*, 2024.

Kyle Hsu, Moo Jin Kim, Rafael Rafailov, Jiajun Wu, and Chelsea Finn. Vision-based manipulators need to also see from their hands. In *ICLR*. OpenReview.net, 2022.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Pu Hua, Minghuan Liu, Annabella Macaluso, Yunfeng Lin, Weinan Zhang, Huazhe Xu, and Lirui Wang. Gensim2: Scaling robot data generation with multi-modal and reasoning llms. *arXiv preprint arXiv:2410.03645*, 2024.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier J. Hénaff, Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver IO: A general architecture for structured inputs & outputs. In *International Conference on Learning Representations*, 2022.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. BC-Z: zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pp. 991–1002. PMLR, 2021.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In *International Conference on Machine Learning*, pp. 9902–9915. PMLR, 2022.

Yueru Jia, Jiaming Liu, Sixiang Chen, Chenyang Gu, Zhilue Wang, Longzan Luo, Lily Lee, Pengwei Wang, Zhongyuan Wang, Renrui Zhang, et al. Lift3d foundation policy: Lifting 2d large-scale pretrained models for robust 3d robotic manipulation. *arXiv preprint arXiv:2411.18623*, 2024.

Zhenyu Jiang, Yuqi Xie, Kevin Lin, Zhenjia Xu, Weikang Wan, Ajay Mandlekar, Linxi Fan, and Yuke Zhu. Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning. *arXiv preprint arXiv:2410.24185*, 2024.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Simar Kareer, Dhruv Patel, Ryan Punamiya, Pranay Mathur, Shuo Cheng, Chen Wang, Judy Hoffman, and Danfei Xu. Egomimic: Scaling imitation learning via egocentric video. *arXiv preprint arXiv:2410.24221*, 2024.

Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*, 2020.

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*, 2024.

Heecheol Kim, Yoshiyuki Ohmura, Akihiko Nagakubo, and Yasuo Kuniyoshi. Training robots without robots: deep imitation learning for master-to-robot policy transfer. *IEEE Robotics and Automation Letters*, 8(5):2906–2913, 2023a.

Moo Jin Kim, Jiajun Wu, and Chelsea Finn. Giving robots a hand: Learning generalizable manipulation with eye-in-hand human video demonstrations. *arXiv preprint arXiv:2307.05959*, 2023b.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, 37(6):222:1–222:11, 2018.

Fanqi Lin, Yingdong Hu, Pingyue Sheng, Chuan Wen, Jiacheng You, and Yang Gao. Data scaling laws in imitation learning for robotic manipulation. *arXiv preprint arXiv:2410.18647*, 2024a.

Xingyu Lin, John So, Sashwat Mahalingam, Fangchen Liu, and Pieter Abbeel. Spawnnet: Learning generalizable visuomotor skills from pre-trained network. In *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*, pp. 4781–4787. IEEE, 2024b.

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. *arXiv preprint arXiv:2410.07864*, 2024a.

Wenhai Liu, Junbo Wang, Yiming Wang, Weiming Wang, and Cewu Lu. Forcemimic: Force-centric imitation learning with force-motion capture system for contact-rich manipulation. *arXiv preprint arXiv:2410.07554*, 2024b.

Shengcheng Luo, Quanquan Peng, Jun Lv, Kaiwen Hong, Katherine Rose Driggs-Campbell, Cewu Lu, and Yong-Lu Li. Human-agent joint learning for efficient robot manipulation skill acquisition. *arXiv preprint arXiv:2407.00299*, 2024.

Jun Lv, Yunhai Feng, Cheng Zhang, Shuang Zhao, Lin Shao, and Cewu Lu. SAM-RL: sensing-aware model-based reinforcement learning via differentiable physics-based simulation and rendering. In *Robotics: Science and Systems*, 2023.

Yecheng Jason Ma, Vikash Kumar, Amy Zhang, Osbert Bastani, and Dinesh Jayaraman. LIV: language-image representations and rewards for robotic control. In *International Conference on Machine Learning*, pp. 23301–23320. PMLR, 2023a.

Yecheng Jason Ma, Shagun Sodhani, Dinesh Jayaraman, Osbert Bastani, Vikash Kumar, and Amy Zhang. VIP: towards universal visual reward and representation via value-implicit pre-training. In *International Conference on Learning Representations*, 2023b.

Arjun Majumdar, Karmesh Yadav, Sergio Arnaud, Yecheng Jason Ma, Claire Chen, Sneha Silwal, Aryan Jain, Vincent-Pierre Berges, Pieter Abbeel, Jitendra Malik, et al. Where are we in the search for an artificial visual cortex for embodied intelligence? In *ICRA 2023 Workshop on Pretraining for Robotics*, 2023.

Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj S. Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1820–1864. PMLR, 2023.

Robert McCarthy, Daniel CH Tan, Dominik Schmidt, Fernando Acero, Nathan Herr, Yilun Du, Thomas G Thuruthel, and Zhibin Li. Towards generalist robot learning from internet video: A survey. *arXiv preprint arXiv:2404.19664*, 2024.

Yao Mu, Tianxing Chen, Shijia Peng, Zanxin Chen, Zeyu Gao, Yude Zou, Lunkai Lin, Zhiqiang Xie, and Ping Luo. Robotwin: Dual-arm robot benchmark with generative digital twins. *arXiv preprint arXiv:2409.02920*, 2024.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *Conference on Robot Learning*, pp. 892–909. PMLR, 2022.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Charles Xu, Jianlan Luo, Tobias Kreiman, 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 *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *Trans. Mach. Learn. Res.*, 2024, 2024.

Georgios Papagiannis, Norman Di Palo, Pietro Vitiello, and Edward Johns. R+ x: Retrieval and execution from everyday human videos. *arXiv preprint arXiv:2407.12957*, 2024.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Dean Pomerleau. ALVINN: an autonomous land vehicle in a neural network. In David S. Touretzky (ed.), *Advances in Neural Information Processing Systems 1, [NIPS Conference, Denver, Colorado, USA, 1988]*, pp. 305–313. Morgan Kaufmann, 1988.

Jianing Qian, Anastasios Panagopoulos, and Dinesh Jayaraman. Recasting generic pretrained vision transformers as object-centric scene encoders for manipulation policies. In *IEEE International Conference on Robotics and Automation, ICRA 2024, Yokohama, Japan, May 13-17, 2024*, pp. 17544–17552. IEEE, 2024.

Yuzhe Qin, Yueh-Hua Wu, Shaowei Liu, Hanwen Jiang, Ruihan Yang, Yang Fu, and Xiaolong Wang. Dexmv: Imitation learning for dexterous manipulation from human videos. In *European Conference on Computer Vision*, pp. 570–587. Springer, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *Conference on Robot Learning*, pp. 416–426. PMLR, 2022.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022.

Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, et al. Zeronvs: Zero-shot 360-degree view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9420–9429, 2024.

Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4104–4113, 2016.

Mingyo Seo, H Andy Park, Shenli Yuan, Yuke Zhu, and Luis Sentis. Legato: Cross-embodiment imitation using a grasping tool. *arXiv preprint arXiv:2411.03682*, 2024.

Nur Muhammad Mahi Shafiullah, Anant Rai, Haritheja Etukuru, Yiqian Liu, Ishan Misra, Soumith Chintala, and Lerrel Pinto. On bringing robots home. *arXiv preprint arXiv:2311.16098*, 2023.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In Karen Liu, Dana Kulic, and Jeffrey Ichnowski (eds.), *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, volume 205 of *Proceedings of Machine Learning Research*, pp. 785–799. PMLR, 2022.

Laura M. Smith, Nikita Dhawan, Marvin Zhang, Pieter Abbeel, and Sergey Levine. AVID: learning multi-stage tasks via pixel-level translation of human videos. In *Robotics: Science and Systems*, 2020.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *The International Conference on Learning Representations*, 2021.

Mohan Kumar Srirama, Sudeep Dasari, Shikhar Bahl, and Abhinav Gupta. Hrp: Human affordances for robotic pre-training. In *Proceedings of Robotics: Science and Systems*, 2024.

Stephen Tian, Blake Wulfe, Kyle Sargent, Katherine Liu, Sergey Zakharov, Vitor Guizilini, and Jiajun Wu. View-invariant policy learning via zero-shot novel view synthesis. In *Conference on Robot Learning (CoRL)*, 2024.

Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *European Conference on Computer Vision*, pp. 313–331. Springer, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

Vitalis Vosylius and Edward Johns. Instant policy: In-context imitation learning via graph diffusion. *arXiv preprint arXiv:2411.12633*, 2024.

Homer Rich Walke, Kevin Black, Tony Z. Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, Abraham Lee, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata V2: A dataset for robot learning at scale. In *Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1723–1736. PMLR, 2023.

Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. In *Conference on Robot Learning (CoRL)*, pp. 201–221, 2023.

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*, 2024a.

Chenxi Wang, Hao-Shu Fang, Minghao Gou, Hongjie Fang, Jin Gao, and Cewu Lu. Graspness discovery in clutters for fast and accurate grasp detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15964–15973, 2021.

Chenxi Wang, Hongjie Fang, Hao-Shu Fang, and Cewu Lu. Rise: 3d perception makes real-world robot imitation simple and effective. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2870–2877, 2024b.

Jun Wang, Chun-Cheng Chang, Jiafei Duan, Dieter Fox, and Ranjay Krishna. Eve: Enabling anyone to train robots using augmented reality. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–13, 2024c.

Jun Wang, Yuzhe Qin, Kaiming Kuang, Yigit Korkmaz, Akhilan Gurumoorthy, Hao Su, and Xiaolong Wang. Cyberdemo: Augmenting simulated human demonstration for real-world dexterous manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17952–17963. IEEE, 2024d.

Lirui Wang, Xinlei Chen, Jialiang Zhao, and Kaiming He. Scaling proprioceptive-visual learning with heterogeneous pre-trained transformers. *arXiv preprint arXiv:2409.20537*, 2024e.

Lirui Wang, Yiyang Ling, Zhecheng Yuan, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, and Xiaolong Wang. Gensim: Generating robotic simulation tasks via large language models. In *International Conference on Learning Representations*, 2024f.

Yufei Wang, Zhou Xian, Feng Chen, Tsun-Hsuan Wang, Yian Wang, Katerina Fragkiadaki, Zackory Erickson, David Held, and Chuang Gan. Robogen: Towards unleashing infinite data for automated robot learning via generative simulation. In *International Conference on Machine Learning*, 2024g.

Chuan Wen, Xingyu Lin, John So, Kai Chen, Qi Dou, Yang Gao, and Pieter Abbeel. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *International Conference on Learning Representations*, 2024a.

Kun Wu, Chengkai Hou, Jiaming Liu, Zhengping Che, Xiaozhu Ju, Zhuqin Yang, Meng Li, Yinuo Zhao, Zhiyuan Xu, Guang Yang, et al. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. *arXiv preprint arXiv:2412.13877*, 2024b.

Shangning Xia, Hongjie Fang, Hao-Shu Fang, and Cewu Lu. Cage: Causal attention enables data-efficient generalizable robotic manipulation. *arXiv preprint arXiv:2410.14974*, 2024.

Haoyu Xiong, Quanzhou Li, Yun-Chun Chen, Homanga Bharadhwaj, Samarth Sinha, and Animesh Garg. Learning by watching: Physical imitation of manipulation skills from human videos. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7827–7834. IEEE, 2021.

Mengda Xu, Zhenjia Xu, Yinghao Xu, Cheng Chi, Gordon Wetzstein, Manuela Veloso, and Shuran Song. Flow as the cross-domain manipulation interface. *arXiv preprint arXiv:2407.15208*, 2024.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, et al. Latent action pretraining from videos. *arXiv preprint arXiv:2410.11758*, 2024.

Sarah Young, Dhiraj Gandhi, Shubham Tulsiani, Abhinav Gupta, Pieter Abbeel, and Lerrel Pinto. Visual imitation made easy. In *Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pp. 1992–2005. PMLR, 2020.

Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

Jia Zeng, Qingwen Bu, Bangjun Wang, Wenke Xia, Li Chen, Hao Dong, Haoming Song, Dong Wang, Di Hu, Ping Luo, et al. Learning manipulation by predicting interaction. In *Proceedings of Robotics: Science and Systems*, 2024.

Junjie Zhang, Chenjia Bai, Haoran He, Zhigang Wang, Bin Zhao, Xiu Li, and Xuelong Li. SAM-E: leveraging visual foundation model with sequence imitation for embodied manipulation. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024a.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3836–3847, 2023a.

Tong Zhang, Yingdong Hu, Hanchen Cui, Hang Zhao, and Yang Gao. A universal semantic-geometric representation for robotic manipulation. In *CoRL*, volume 229 of *Proceedings of Machine Learning Research*, pp. 3342–3363. PMLR, 2023b.

Tong Zhang, Yingdong Hu, Jiacheng You, and Yang Gao. Leveraging locality to boost sample efficiency in robotic manipulation. *arXiv preprint arXiv:2406.10615*, 2024b.

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

Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018.

Shangchen Zhou, Chongyi Li, Kelvin CK Chan, and Chen Change Loy. Propainter: Improving propagation and transformer for video inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10477–10486, 2023.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5745–5753, 2019.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong T. Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: vision-language-action models transfer web knowledge to robotic control. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 2023.

A Appendix

A.1 AirExo-2 Hardware Design

To collect in-the-wild data that is easy to transform into pseudo robotic data, a highly precise data collection system is essential to ensure demonstration quality for downstream policy learning. From the hardware perspective, AirExo (Fang et al., 2024b) has several key limitations:

- **H1**. Most of its components are 3D-printed using polylactic acid (PLA), leading to low rigidity and susceptibility to structural deformation.
- **H2**. Unlike typical robots with constrained ranges, its joints can rotate beyond 360°, potentially reaching positions the robot cannot achieve.
- **H3**. Although portable, inevitable body movements during operations can cause its base to shift, leading to inaccurate action recording.
- **H4**. Its gripper lacks smooth control, leading to potential jamming under clamping forces.
- **H5**. Its shaft connects directly to the encoder, with wires routed along its side. Joint movement can cause constant friction and stretching, risking wire breakage or short circuits over long-time use.

During demonstration collection using teleoperation, most of the above issues (except **H4**) might not significantly impact data acquisition, as the human operator can adjust their actions based on the movements of the robotic arms. However, during the in-the-wild demonstration collection, these drawbacks can substantially affect motion capture accuracy and may result in invalid or unusable demonstration data.

The hardware design of ***AirExo*-2** is primarily driven by the limitations mentioned above. To ensure seamless integration with the learning process from in-the-wild demonstrations, we design the exoskeleton to match the dimensions of the robotic arm in a 1:1 ratio. This design choice helps avoid unnecessary obstacles in direct learning from the collected demonstrations. The key hardware designs are outlined as follows.

Enhanced Overall Structural Rigidity (H1). In AirExo, the links connecting two consecutive joints are the most prone to deformation. In ***AirExo*-2**, we replace the 3D-printed parts with 20x20 European standard aluminum profiles, providing significantly higher strength at a very low cost. For the joints, the outer shell is 3D-printed using PLA-CF, a carbon fiber-reinforced PLA material with higher hardness. Inside the joint, larger bearings are used to further enhance structural rigidity. Together with the improved links, the hardware upgrade significantly increases the overall structural rigidity of ***AirExo*-2**, making the exoskeleton more durable, and thereby improving data collection accuracy.

Hollow Rotating Disc and Side-Mounted Encoder (H5). As shown in Fig. 9, this design features an encoder mounted on the side of the joint, which uses a gear mechanism to translate the rotational angle of the rotating disc into encoder readings. The hollow disc design allows wires to pass through, preventing them from stretching during rotation and thereby extending their lifespan. Additionally,

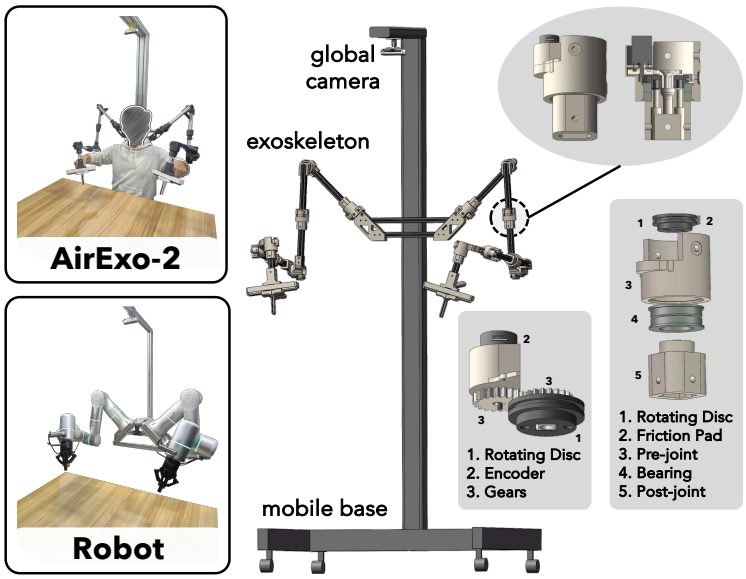


Figure 9: Hardware Design of *AirExo*-2. The ***AirExo*-2** demonstration collection platform consists of a mobile base and a dual-arm exoskeleton. Global cameras can be mounted on top of the platform to capture visual observations during data collection. The detailed joint structure is shown on the right side, featuring two key designs: hollow rotating disc and side-mounted encoder; joint with angle limit and adjustable friction.

the side-mounted encoder simplifies maintenance, enabling easy debugging and replacement without the need to disassemble the joint.

Joint with Angle Limit and Adjustable Friction (H2). This structure consists of a rotating disc with a grooved track and a friction pad that can be embedded into the track. Together, they allow for adjustment of the limiting angle, ensuring that the motion range of the ***AirExo*-2** joint exactly aligns with the corresponding joint range of the robot. As illustrated in Fig. 9, the rotating disc, pre-joint, and post-joint are connected through bearings, allowing the rotational motion of the joint to be directly transmitted to the track. For demonstration collection, excessive or insufficient friction in the rotation of the joints is undesirable. Hence, the friction force of the joint in ***AirExo*-2** can be adjusted by turning the screw on the outer shell, which compresses the friction pad. This design ensures optimal friction for comfortable and accurate data collection.

Smooth Gripper Control (H4). Following (Chi et al., 2024; Liu et al., 2024b), the gripper of AirExo-2 incorporates a linear guide, with the fingers mounted on a sliding block. This design allows for smoother opening and closing of the gripper, ensuring it operates seamlessly even under significant clamping forces without any stalling.

Mobile Data Collection Platform (H3). Portability is crucial for in-the-wild data collection. However, to address the issue of base movement caused by the body motion of the operator, we mount ***AirExo*-2** on a mobile aluminum profile stand, as shown in Fig. 9. This setup ensures stability of the base during demonstration collection while maintaining the flexibility needed for mobility, enabling large-scale demonstration collection in real-world environments. An Intel RealSense D415 camera is set up on the top of the mobile platform to capture global observations. We also designed two optional camera mounts (though not used in this paper) for the future integration of in-hand cameras on the top of both grippers.

A.2 AirExo-2 Calibration

The ***AirExo*-2** system requires two types of calibration simultaneously: (1) aligning the zero positions of each joint with the corresponding robot joint, and (2) determining the transformation between the global camera and the ***AirExo*-2** base. To address these challenges, we propose a two-stage calibration process.

Initial Calibration. For initial calibration, the former calibration can be achieved by manually adjusting the joints to approximate the zero position using specialized 3D-printed tools and reading the encoder values, obtaining $\{\tilde{q}^{\text{left}}_{\text{calib}}, \tilde{q}^{\text{right}}_{\text{calib}}\}$. The latter calibration can be done by attaching an ArUco calibration marker board with a known position on the base $\mathbf{T}^{\text{base}}_{\text{marker}}$ and performing optical calibration using the OpenCV library (Bradski, 2000), obtaining $\mathbf{T}^{\text{camera}}_{\text{marker}}$. Thus, the transformation can calculated as

$$\left[\tilde{\mathbf{t}}^{\text{camera}}_{\text{base}} \mid \tilde{\mathbf{r}}^{\text{camera}}_{\text{base}}\right] \overset{\text{def}}{=} \tilde{\mathbf{T}}^{\text{camera}}_{\text{base}} = \mathbf{T}^{\text{camera}}_{\text{marker}} \left(\tilde{\mathbf{T}}^{\text{base}}_{\text{marker}}\right)^{-1} \tag{6}$$

However, this approach introduces errors due to human observation, calibration board misalignment, and optical inaccuracies. In a chained system like ***AirExo-2***, these errors may propagate and amplify across joints, leading to significant cumulative inaccuracies in the end-effector pose. Therefore, fine-grained calibration is essential to ensure precise and consistent alignment between ***AirExo-2*** and the camera frames during demonstration collection.

Calibration via Differentiable Rendering. In the second stage, inspired by prior works (Chen et al., 2023a; Hong et al., 2024; Lv et al., 2023), we use differentiable rendering (Kato et al., 2020) to refine the initial calibration. Training samples are obtained from a single human *play* trajectory with ***AirExo-2***. Using the joint states and calibration parameters, we render the system mask and depth via a differentiable rendering engine (Li et al., 2018). Calibration parameters are optimized by minimizing discrepancies between the rendered and annotated system masks, as well as between the rendered and observed depths. Pseudo-ground-truth masks are manually annotated with SAM-2 (Ravi et al., 2024). This iterative refinement compensates for errors that accumulate across joints, ultimately improving the overall accuracy.

Specifically, we define p, the calibration parameters to be optimized through differentiable rendering, as

$$p \overset{\text{def}}{=} \{\Delta\mathbf{t}^{\text{camera}}_{\text{base}}, \mathbf{r}^{\text{camera}}_{\text{base}}, \Delta q^{\text{left}}_{\text{calib}}, \Delta q^{\text{right}}_{\text{calib}}\} \tag{7}$$

where parameters, except for the base-to-camera rotation, are represented as deltas relative to the initial calibration results. The base-to-camera rotation is expressed in a 6D format (Zhou et al., 2019). Thus, the final calibration results can be calculated as

$$\mathbf{T}^{\text{camera}}_{\text{base}} = \left[\tilde{\mathbf{t}}^{\text{camera}}_{\text{base}} + \Delta\mathbf{t}^{\text{camera}}_{\text{base}} \mid \mathbf{r}^{\text{camera}}_{\text{base}}\right], \tag{8}$$

$$q^{\text{type}}_{\text{calib}} = \tilde{q}^{\text{type}}_{\text{calib}} + \Delta q^{\text{type}}_{\text{calib}}, \quad \text{type} \in [\text{left}, \text{right}], \tag{9}$$

and the initial parameter values are set as

$$p_0 = \{\mathbf{0}, \tilde{r}^{\text{camera}}_{\text{base}}, \mathbf{0}, \mathbf{0}\} \tag{10}$$

For the optimization process, we first record a single in-the-wild *play* trajectory, in which the human operator uses the ***AirExo-2*** to adopt various poses. During trajectory recording, ensure that all parts of the ***AirExo-2*** remain above the human hands and arms from the camera's perspective. After data collection, we sample approximately 40 image-joint pairs from the trajectory, denoted as $\{I_i, d_i, q^{\text{left}}_i, q^{\text{right}}_i\}_{i=1}^{N_c}$, where N_c represents the total number of training samples for calibration, and I_i and d_i are the RGB and depth images of the i-th sample, respectively. Subsequently, we utilize SAM-2 (Ravi et al., 2024) to annotate the ***AirExo-2*** mask M^a_i and the depth mask $M^d_i \subseteq M^a_i$ for each sample i, as shown in Fig. 10. The first mask provides supervision for the rendered ***AirExo-2*** mask, while the second mask is used to select the valid depth information that serves as the supervision signal.

The differentiable rendering engine Redner (Li et al., 2018) is employed to render the ***AirExo-2*** mask $\hat{M}^a_i$ and ***AirExo-2*** depth $\hat{d}_i$ using the calibration results and joint information:

$$\hat{M}^a_i, \hat{d}_i = \mathscr{R}(p; q^{\text{left}}_i, q^{\text{right}}_i, \tilde{T}^{\text{camera}}_{\text{base}}, \tilde{q}^{\text{left}}, \tilde{q}^{\text{right}}), \tag{11}$$

where the rendering engine $\mathscr{R}(p; \cdots)$ computes the gradients of the calibration parameters p during the rendering process, and $\tilde{T}^{\text{camera}}_{\text{base}}, \tilde{q}^{\text{left}}, \tilde{q}^{\text{right}}$ represent the initial calibration results.

The rendered ***AirExo-2*** mask $\hat{M}^a_i$ is supervised by the human-annotated pseudo-***AirExo-2*** mask M^a_i, and the rendered ***AirExo-2*** depth $\hat{d}_i$ is supervised by the camera depth d_i within the region of the

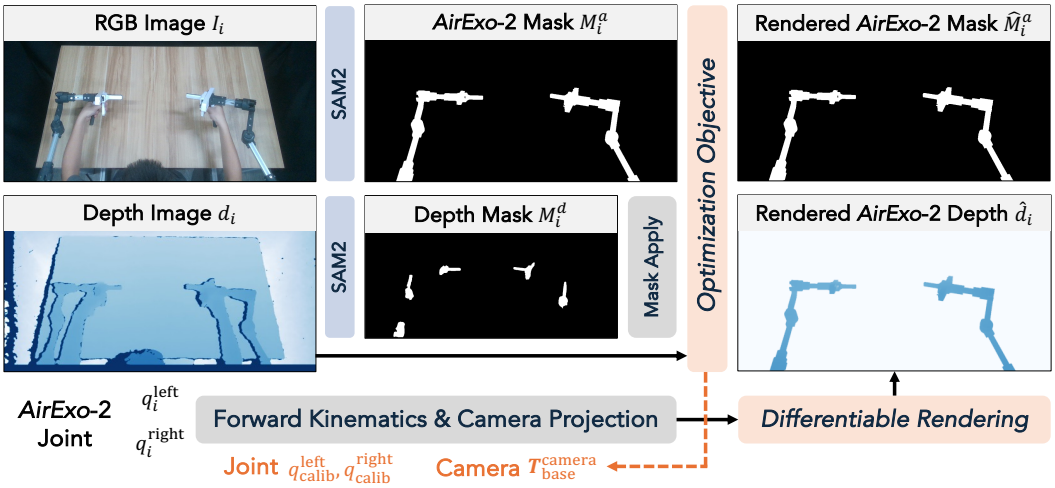


Figure 10: Calibration via Differentiable Rendering. The parameters in orange denote the calibration parameters to be optimized via differentiable rendering.

human-annotated depth mask M_i^d. The depth mask ensures that only accurate depth information contributes to the loss. Thus, the objective can be written as:

$$L = \frac{1}{N_c} \sum_{i=1}^{N_c} \left(\beta \cdot \left\| M_i^a - \hat{M}_i^a \right\|^2 + \left\| d_i - \hat{d}_i \right\|^2 \circ M_i^d \right), \tag{12}$$

where β represents the weighting coefficient, and $\circ$ denotes the mask-apply operation. In practice, we set $\beta = 5$, use $N_c = 40$ samples for optimization, and employ the Adam optimizer (Kingma, 2014) with a learning rate of 10^{-4} for 1000 iterations to fine-tune the calibration results.

A.3 Calibration Analysis

Apart from the two-stage calibration process described in Appendix A.2, we implement several calibration alternatives, including (1) **initial calibration**, which uses the calibration results from the first stage without further fine-tuning; (2) **human annotation**, where a human operator utilizes a real-time GUI program to manually adjust the calibration parameters to align the rendered ***AirExo-2*** contour with the visually observed contour from camera frames; and (3) **two-stage calibration (mask only)**, which fine-tunes the calibration results by using only mask differences as supervision.

Method	Difference ↓	
	Mask (%)	**Depth** (mm)
Initial Calibration	$1.71_{\pm 0.37}$	$21.6_{\pm 5.2}$
Human Annotation	$2.31_{\pm 0.31}$	$31.2_{\pm 6.4}$
Two-Stage Calibration (mask only)	$1.10_{\pm 0.26}$	$17.6_{\pm 4.1}$
Two-Stage Calibration (mask + depth)	$\mathbf{0.78}_{\pm 0.25}$	$\mathbf{14.0}_{\pm 2.9}$

Table 6: Calibration Analysis Results. Using our proposed two-stage calibration, we achieve higher accuracy than both initial calibration and human annotations. Including depth as additional supervision also helps the optimization process convergence.

Our two-stage calibration process achieves more accurate results compared to other alternatives. As reported in Tab. 6, our two-stage calibration process achieves the lowest error rates, with a 0.78% mask difference and 14.0 mm depth difference, yielding more precise calibration. It's worth noticing that the depth difference here does not fully represent action accuracy because commercial depth sensors can produce noisy depth maps. Please refer to §6.2 for more details about action accuracy. Interestingly, human annotation performs even worse than initial calibration, mainly because annotators can only rely on 2D visual information to adjust calibration parameters. This limitation

makes it difficult to accurately estimate depth information, leading to larger errors. Conversely, our two-stage calibration process explicitly models 3D information via differentiable rendering, offering a more reliable and precise solution for calibrating the ***AirExo-2*** system.

A.4 Demonstration Adaptor

Semi-Automatic SAM-2 Annotations. In the image adaptor, we initially annotate the hand mask (and, if visible, the head mask) manually using SAM-2 (Ravi et al., 2024). However, after a few annotations, we can fine-tune SAM-2 on the human-annotated samples, enabling automated labeling. This significantly reduces human effort and streamlines the demonstration adaptor process, making it nearly fully automated.

ControlNet Training. We train a ControlNet (Zhang et al., 2023a) based on the Stable Diffusion 1.5 (Rombach et al., 2022) model to generate photo-realistic robot images from rendered robot images. To collect training samples, we use teleoperation to gather a small amount of *play* data, where the robot is teleoperated to move randomly within an empty workspace while recording RGB-D images and corresponding joint states. This ensures a diverse dataset of robot arm configurations, free from occlusions or distractions.

Notably, these training samples are ***platform-specific*** but ***task-invariant***, meaning they only need to be collected once per robot platform, and the trained ControlNet can be used across all tasks. This also opens up the possibility of directly transforming our in-the-wild demonstrations to other robotic arms without the need to design new exoskeletons that match specific robots.

For training, we use a batch size of 88 and a learning rate of 10^{-5}, while keeping other hyperparameters at their default settings. We use 50 DDPM sampling steps (Ho et al., 2020a) with a guidance scale of 9.0 (Ho & Salimans, 2022). The prompt for generating robot images for our robot platform is set to:

robotic arms, dual arm, industrial robotic manipulator, metallic silver color, mechanical joints, precise mechanical details, gripper end effector, high-quality photo, photorealistic, clear and sharp details

A.5 RISE-2 Implementation

Data Processing. The color image is resized to 448×252 for DINOv2 backbone (Oquab et al., 2024) and 640×360 for ResNet-18 backbone (He et al., 2016). The depth image is resized to 640×360 before creating the point cloud. The camera intrinsics are adjusted accordingly. Both the point clouds and actions are in the camera coordinate system. The point cloud is down-sampled with a voxel size of 5mm. For the data collected with teleoperation, we crop the point clouds using the range of $x \in$[-0.7m, 0.7m], $y \in$[-0.3m, 0.55m] and $z \in$[0.9m, 1.55m]. For the data collected with ***AirExo-2***, we crop the point clouds using the range of $x \in$[-0.7m, 0.7m], $y \in$[-0.3m, 0.45m] and $z \in$[0.75m, 1.4m].

The robot trajectories are sampled using differences of translation, rotation and gripper width to remove redundant actions. For the action at two adjacent timesteps, if all the differences are less than the thresholds, only the first action is retained. The threshold for translation and gripper width is 5mm and the rotation threshold is $\pi/24$.

Network. The sparse encoder adopts a ResNet-like architecture built upon MinkowskiEngine (Choy et al., 2019). The dense encoder adopts DINOv2-base (Oquab et al., 2024) as the 2D backbone with the output channel of 128. In the spatial aligner, we use $M = 3$ for feature alignment. The aligned features are fused by shared MLPs with the size of (256, 256, 256), and then fed into another sparse network. The two sparse networks are detailed in Tab. 7. The transformer in action generator contains 4 blocks, in which we set $d_{\text{model}} = 512$ and $d_{\text{ff}} = 2048$. The channel number of the readout token is 512. The diffusion head adopts a CNN implementation (Chi et al., 2023) with 100 denoising iterations in training and 20 iterations in inference. The output action horizon used in experiments is 20.

Training. ***RISE-2*** is trained on 4 Nvidia A100 GPUs. The batch size is 240, the initial learning rate is 3e-4, and the warmup step is 2000. We employ a cosine scheduler to adjust the learning

Layer Name	Sparse Encoder	Spatial Aligner
init_conv	$k=[3,3,3]$, $c=32$, $d=1$, $s=1$ 2x mean pooling	-
conv1	$k=[3,3,3]$, $c=32$, $d=1$, $s=1$	$k=[3,3,3]$, $c=256$, $d=4$, $s=4$
conv2	$k=[3,3,3]$, $c=64$, $d=2$, $s=1$	$k=[3,3,3]$, $c=256$, $d=1$, $s=2$
conv3	$k=[3,3,3]$, $c=128$, $d=4$, $s=1$	$k=[3,3,3]$, $c=512$, $d=1$, $s=2$
conv4	$k=[3,3,3]$, $c=128$, $d=8$, $s=2$	$k=[3,3,3]$, $c=512$, $d=1$, $s=2$
final_conv	$k=[1,1,1]$, $c=128$, $d=1$, $s=1$	$k=[1,1,1]$, $c=512$, $d=1$, $s=1$

Table 7: Sparse Convolutional Network Parameters of the *RISE*-2 Policy. Both sparse encoder and spatial aligner utilize MinkResNet (Choy et al., 2019) for point cloud encoding. k,c,d,s stand for the kernel size, output channel number, dilation and stride in the convolutional layers respectively.

Method	Action Repr.	Success Rate (%) ↑		
		in-domain	background	object
DP (Chi et al., 2023)	relative	37.5	**20.0**	**5.0**
	absolute	**40.0**	12.5	**5.0**
CAGE (Xia et al., 2024)	relative	**72.5**	32.5	35.0
	absolute	65.0	**45.0**	**42.5**

Table 8: Evaluation Results of Different Action Representations on the *Collect Toys* Task. Absolute action representation leads to a more stable performance.

rate during training. 20% of the color images are augmented using a color jitter with (brightness, contrast, saturation, hue) parameters set to (0.4, 0.4, 0.2, 0.1).

A.6 Action Representations

We conduct additional experiments on action representations for the ***Collect Toys*** task, comparing relative and absolute action representations. The results in Tab. 8 show that while relative action representation sometimes yields better results, absolute action representation provides more stable performance, particularly in terms of generalization. Therefore, we use absolute action representations throughout our experiments, except for the case study in §5.2.

A.7 Accuracy Analysis

To evaluate action accuracy, we designed a special evaluation board with three tracks, as illustrated in Fig. 11. Each track has fixed holes spaced 2 cm apart. We created custom connectors for both the ***AirExo*-2** and UMI (Chi et al., 2024) that fit into these fixed holes, allowing us to collect position data. By sequentially placing the connectors into each fixed hole along the track, we can calculate the relative movement distance between two adjacent fixed holes and compare it with the true value (20 mm) to calculate the error.

A.8 Visualization of Sparse Semantic Features

Fig. 12 visualizes the sparse semantic features obtained from the dense encoder by projecting the 2D feature map to 3D form using the reference coordinates. The sparse semantic features are aligned to the input point cloud using weighted spatial interpolation detailed in §4.2.

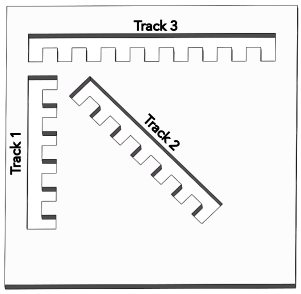


Figure 11: The Designed Evaluation Board for Accuracy Analysis.

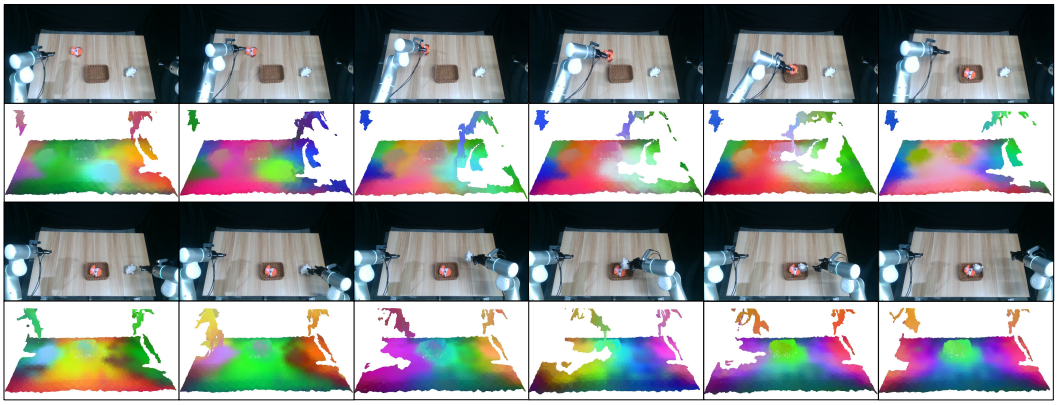


Figure 12: Visualization of Sparse Semantic Features. The colors are obtained by performing PCA on the features. The original sparse semantic features are aligned to the input point cloud using weighted spatial interpolation function in the spatial aligner for clearer visualization.

Although the 2D feature map output by the dense encoder is in low resolution (32×18), we still observe clear and distinguishable continuous feature variations on the aligned features, where the targets at the current step can be easily identified from the entire scene. Such characteristic ensures precise feature fusion in the spatial domain. Additionally, we find that the features from DINOv2 change significantly as the task progresses, enabling the model to clearly understand the global state at the current time.