2026 IEEE International Conference on Robotics and Automation (ICRA)
June 1-5, 2026. Vienna, Austria

Dexterity from Smart Lenses: Multi-Fingered Robot Manipulation with In-the-Wild Human Demonstrations

Irmak Guzey[1,2] Haozhi Qi[2] Julen Urain[2] Changhao Wang[2] Jessica Yin[2]
Krishna Bodduluri[2] Mike Lambeta[2] Lerrel Pinto[1] Akshara Rai[2]
Jitendra Malik[2] Tingfan Wu[2] Akash Sharma[2] Homanga Bharadhwaj[2]

[1] New York University, [2] Meta

aina-robot.github.io

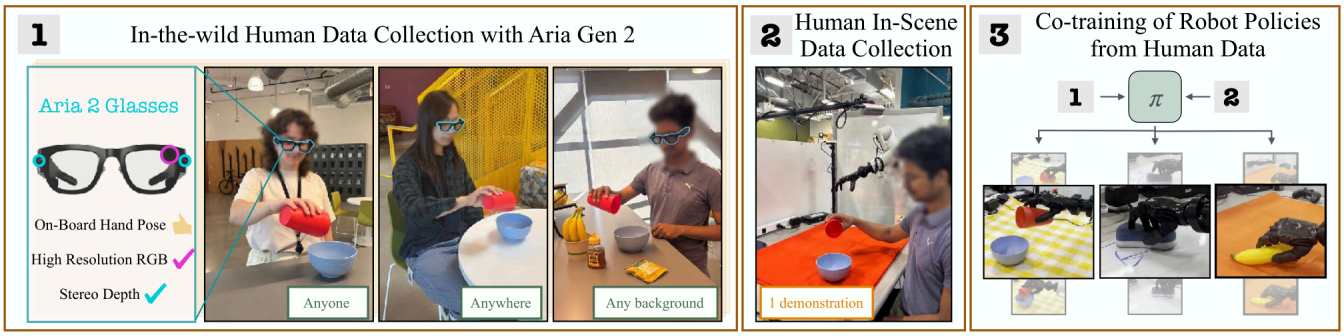

Fig. 1: AINA is a framework for learning multi-fingered policies from in-the-wild human data collected with smart glasses, without requiring any robot data (including online corrections or simulation). The workflow is as follows: a human wears the Aria 2 glasses and collects in-the-wild demonstrations on any surface with arbitrary backgrounds (left), then records a single demonstration in the robot deployment space (middle), after which point-based policies are trained and directly deployed on the robot (right). With an average of just 15 minutes of human video collection effort, AINA is able to train autonomous robot policies.

***Abstract*— Learning multi-fingered robot policies from humans performing daily tasks in natural environments has long been a grand goal in the robotics community. Achieving this would mark significant progress toward generalizable robot manipulation in human environments, as it would reduce the reliance on labor-intensive robot interaction data collection. Despite substantial efforts, progress toward this goal has been bottle-necked by the embodiment gap between humans and robots, as well as by difficulties in extracting relevant contextual and motion cues that enable learning of autonomous policies from in-the-wild human videos. We claim that with simple yet sufficiently powerful hardware for obtaining human data and our proposed framework AINA, we are now one significant step closer to achieving this dream. AINA enables learning multi-fingered policies from in-the-wild data using Aria Gen 2 glasses. These glasses are lightweight and portable, feature a high-resolution RGB camera, provide accurate on-board 3D head and hand poses, and offer a wide stereo view that can be leveraged for depth estimation of the scene. This setup enables the learning of 3D point-based policies for multi-fingered hands that are robust to background changes and can be deployed directly without requiring any robot data (including online corrections, reinforcement learning, or simulation). We compare our framework against prior human-to-robot policy learning approaches, ablate our design choices, and demonstrate results across five everyday manipulation tasks. Robot rollouts can be best viewed on our website: https://aina-robot.github.io.**

Correspondence to irmakguzey@nyu.edu.

I. INTRODUCTION

"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."

— Mark Weiser, 1991

Robots autonomously performing diverse manipulation tasks by watching humans go about their daily lives has been a dream in Artificial Intelligence (AI) for decades. However, this remains challenging due to the embodiment gap between humans and robots, as well as the disparity between human video views and the sensor perspectives of a robot. To truly realize this dream for generalizable dexterous manipulation, we must overcome these challenges with general approaches that can leverage large-scale human video data. Encouragingly, this vision is now closer to reality with development of increasingly more human-like robot embodiments [3], [4] and the potential widespread adoption of wearable devices such as smart glasses, which are lightweight, easy to wear in daily life and equipped with complex sensing capabilities that provide both an egocentric perspective and rich annotations. Building on this promise, we develop an approach to learn dexterous manipulation directly from smart-glass human data, without requiring any additional robot interaction data.

2026 IEEE International Conference on Robotics and Automation (ICRA) | 979-8-3315-8160-2/26/$31.00 ©2026 IEEE | DOI: 10.1109/ICRA57385.2026.11697220

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

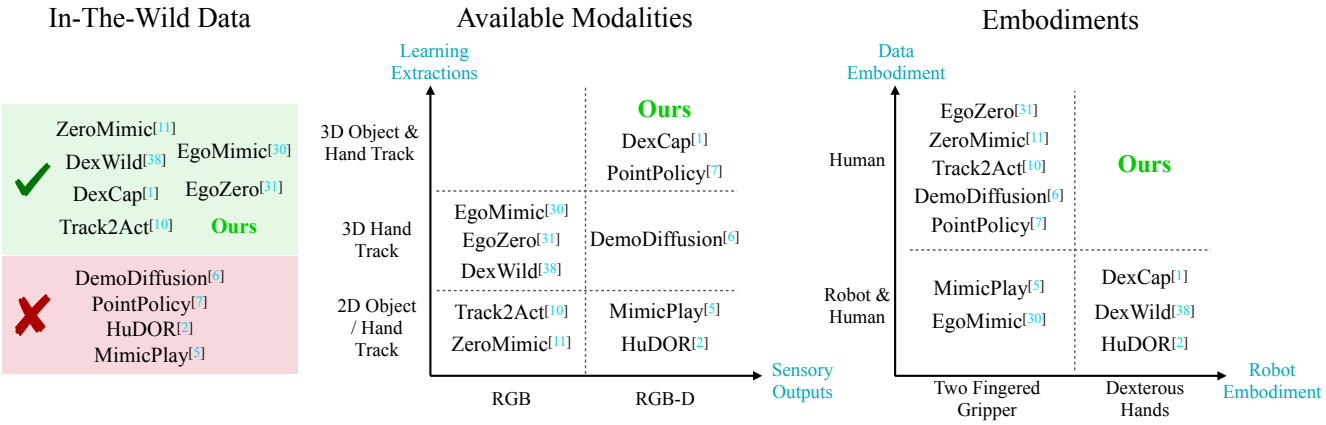

Fig. 3: Comparison of AINA's capabilities with prior human-to-robot learning frameworks. *In-The-Wild* indicates whether data can be easily collected in natural settings outside the lab. *Sensors* describes the sensory outputs available from the data collection devices. *Learning Extractions* specifies which extractions can be utilized with the provided sensors to improve learning. *Data Embodiment* refers to the embodiment of the collected data (robot vs. human). Here, we also count online corrections [1] and reinforcement learning [2] performed on the robot as part of the robot data. *Robot Embodiment* indicates which type of robot embodiment the framework targets (two-fingered gripper vs. multi-fingered hand). In AINA, we choose point-based approaches for their robustness to background variations, enabling robot learning from in-the-wild data for dexterous hands. This is made possible by the advanced sensing capabilities of the Aria Gen 2 glasses, which provide all the necessary 3D extractions. More comparisons is available on our website: https://aina-robot.github.io.

We are, of course, not the first to consider this setting of learning manipulation from human videos. Prior work has attempted to address these challenges by collecting human videos in structured settings, often within the exact scenarios of robot deployment [5]–[7]. However, such approaches are difficult to scale to diverse environments, as they require data collection for each deployment scenario. Other efforts leverage large-scale, in-the-wild web videos [8]–[11], but they have not been successfully deployed on multi-fingered hands, since extracting the necessary annotations—such as reliable 3D hand poses—for learning dexterous policies is far more challenging in these settings. Smart-glasses data, offers the best of both worlds: it preserves scalability by being naturally collected as people go about daily life, while providing high-resolution egocentric imagery, stereo vision for 3D perception, and reliable hand-pose annotations via in-built software [12]. These characteristics make smart-glass data far richer and more robot manipulation-relevant than web video, while avoiding the scalability bottlenecks of lab-constrained data collection.

Thus, leveraging the complex sensing capabilities of smart glasses, in particular of Aria Gen 2, we develop **AINA**: a simple approach for learning a closed-loop dexterous manipulation policy from just human videos. AINA (english. Mirror) refers to mirroring human videos in a robot's context and is based on a simple intuition: By lifting human videos to approximate 4D via hand-keypoint reconstruction, stereo depth estimation, and 3D object pointcloud extraction, we can re-purpose 3D policy learning approaches for learning to predict future hand keypoints, and use the same policy for robot manipulation. By operating in the space of 3D keypoints for the hand, and 3D pointclouds for objects, we minimize the human-robot domain gap when deploying the AINA policy on a dexterous robot hand, while being trained with only human demonstrations.

Concretely, AINA operates as follows: (a) humans wearing smart glasses collect data in arbitrary environments with any background or viewpoint, (b) then they collect a single video demonstration in the robot's environment, and (c) the multi-fingered robot learns policies that generalize across both spatial configurations and object variations. We evaluate AINA on nine tasks and summarize our contributions as follows:

1) AINA is the first framework that learns policies for multi-fingered hands without using any robot data, including no use of simulation (Section III).
2) AINA introduces a novel domain-alignment approach between the robot environment and in-the-wild human demonstrations that leverages a variety of computer vision techniques. This enables the learning and deployment of closed-loop policies directly from in-the-wild human demonstrations.

In Section IV, we show that AINA outperforms existing human-to-robot learning approaches demonstrating the effectiveness of learning manipulation from human videos alone through a simple framework operating on rich sensing from smart glasses. Robot videos are available on our website: https://aina-robot.github.io.

II. Related Works

AINA draws inspiration from extensive research in dexterous manipulation, learning from human videos, and imitation learning. Our aim is to develop a *simple* framework for closed-loop policy learning capable of performing diverse everyday manipulation tasks with dexterous multi-fingered hands. We highlight some of the comparisons with prior works in Fig. 3 and describe them below.

a) Robot Learning with Non-Robot Datasets: Since robot interaction data collection is challenging due to operational constraints [13], [14], thanks to advances in representation learning [15], [16], motion prediction [17], [18], and hand–object reconstruction [19], [20], many approaches

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

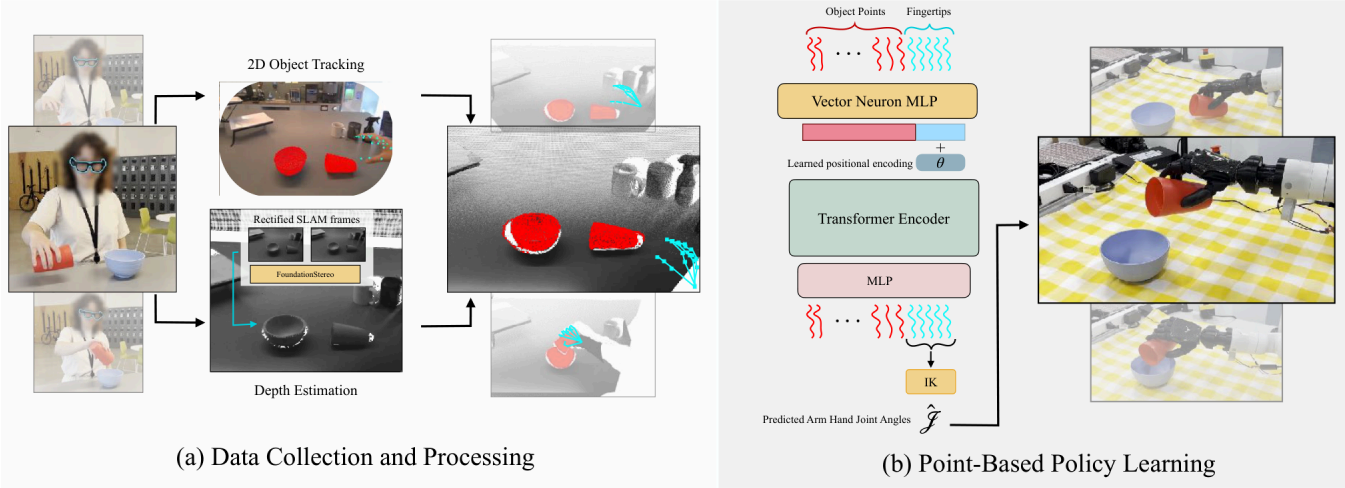

Fig. 4: Illustration of our overall AINA framework. On the left, we show how the data is processed: the human hand pose is extracted directly by the Aria Gen 2 glasses, and stereo depth is estimated from the surrounding SLAM camera frames. This enables the 3D policy learning methods on the right to succeed while remaining robust to background clutter.

now leverage non-robot datasets such as human videos and images. These approaches differ both in the type of human data used—in-domain vs. in-the-wild—and in what is extracted or learned from such data.

In-domain demonstrations, collected in the same environment as deployment, allow rich extractions like 3D hand poses and object points [2], [5]–[7], but require new data per deployment and are thus hard to scale. In contrast, in-the-wild human datasets [21]–[23] support broader generalization, with works focusing on visual backbones [24]–[26] or high-level cues such as hand-object trajectories [10], [11], [27] and affordances [9], [28]. Yet, without reliable low-level signals like 3D hand pose, these methods often sacrifice accuracy or need additional demonstrations during deployment [11].

More recently, smart glasses [29] have simplified data collection [30], [31], enabling richer extractions and better generalization which AINA builds upon. However, most of these works focus on two-finger grippers, where manipulation can be modeled simpler. In AINA, we use Aria Gen 2 glasses [32] for scalable human data collection, but uniquely demonstrate policy learning from purely human demonstrations for dexterous multi-fingered robot hands.

b) Dexterous Manipulation from Human Data: Early research on dexterous manipulation relied on sim-to-real transfer [33], [34] or teleoperation for data collection [13], [14], [35], [36], but these approaches are limited either by sim-to-real gaps or by the extensive human effort required for teleoperation. To address these challenges and reduce dependence on large-scale robot demonstrations, recent work has shifted toward learning from human data. However, leveraging large-scale datasets is harder for multi-fingered hands due to the lack of annotations needed for extracting reliable signals. As a result, most prior works have collected their own human data to train policies. Some collected in-domain human videos and extracted 3D hand poses [2], [37], while others gathered in-the-wild demonstrations using portable custom hardware with multiple cameras and hand-pose estimators [1], [38]. While promising, all of these approaches incorporated some robot data, obtained either through teleoperation [38] or online corrections [1], [2]. Although such robot data can help in complex dexterous tasks—particularly given the absence of force feedback in human demonstrations—in AINA we demonstrate how we can learn to perform everyday manipulation activities with dexterous multi-fingered hands with just offline human videos captured through Aria glasses, without using any external sensors, mocap markers, or exo-skeletons.

c) Policy Architectures for Imitation Learning: Going beyond the standard of 2D image-based policies [24], [39], [40] for imitation, recent works have proposed 3D policy architectures that exploit geometric structure for manipulation [41]–[43], yielding improved generalization to cluttered scenes and complex object interactions. Beyond raw pixels and scene point clouds, some approaches incorporate intermediate object-centric representations such as keypoints or tracks. PointPolicy [7] learns manipulation policies from 3D hand and object keypoints, while Track2Act [10] predicts future dense object tracks from video datasets and trains track-conditioned policies. These object-centric methods highlight the benefits of embodiment-agnostic cues for bridging human and robot domains. Building on this insight, our proposed approach, AINA extends 3D imitation learning frameworks by extracting hand keypoints and 3D object flow from human videos, enabling policies that generalize across embodiments and leverages (non-robot) human data for dexterous manipulation.

III. METHOD

A. Overview

AINA is a framework for learning closed-loop policies from in-the-wild human demonstrations collected with Aria Gen 2 glasses, without requiring any robot data. Our framework consists of three high-level steps: (a) a human collects

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

in-the-wild video demonstrations on arbitrary surfaces using the Aria Gen 2 glasses, along with a single in-scene video in the robot's environment; (b) the dataset is processed to extract 3D object tracks and hand fingertip points, which are then aligned to establish a uniform reference frame with the robot; and (c) point-based policies are trained and deployed on a single-arm hand robot system. We describe the overall structure of our framework in Fig. 4 and describe the assumptions, challenges, and details in this section.

a) ***Assumptions:*** In AINA, our methodology is guided by two key assumptions: (a) access to a calibrated scene to ensure a uniform operational space. For this, we perform hand–eye calibration to compute the extrinsic matrix of cameras on the robot setup with respect to the robot base. This process is straightforward, performed only once, and takes approximately 5–10 minutes during the initial setup of the robot. (b) access to a single in-scene demonstration along with multiple in-the-wild demonstrations. Both types of demonstration are collected by humans (without using the robot). The in-scene demonstration takes less than a minute to collect, while the in-the-wild demonstrations take about 10 minutes in total for 50 demos per task.

b) ***Challenges:*** Unlike prior works that artificially constrain hand motions to be robot-like [44] or require additional alignment hardware such as ArUco markers [31], our approach considers in-the-wild human videos as *natural* interactions. Our method does not require prior knowledge of the distance to manipulated objects, and it places no restrictions on the hand motions of the data collectors [7]. These relaxations introduce significant challenges, as the hand motions are less structured. In subsequent subsections, we describe how we address these challenges.

B. Collecting and Processing Smart Glass Data

1) Data Collection: AINA uses Project Aria Gen 2 [32] glasses to collect in-the-wild human demonstrations. The glasses are equipped with a front-facing RGB camera, four SLAM cameras positioned around the frame, and multiple IMUs. These sensors enable real-time estimation of the user's head pose as well as left and right hand poses [12]. The head pose is defined with respect to a world frame arbitrarily assigned at initialization. This world frame is initialized using the gravity vector measured by the IMUs [29], [32], ensuring that its z-axis is aligned with gravity. For each task, we collect 50 in-the-wild demonstrations using these glasses and record the camera streams along with head and hand pose estimates at 15 Hz.

During data collection, we do not assume any specific height for the surfaces where humans manipulate the objects. As a result, we need to ground in-the-wild demonstrations within the robot's scene. To address this, we collect a single *in-scene* demonstration using the RGB-D cameras in the robot environment. We estimate the hand pose in 2D from both camera views using Hamer [45], [46], and then triangulate these estimates to obtain the 3D pose [6], [7]. We collect this in-scene demonstration at 10 Hz.

2) Processing and Object Tracking: AINA uses object point clouds as observations during policy learning. This representation makes the observations invariant to background changes and visual differences between humans and robots. To obtain these object point clouds, we leverage off-the-shelf computer vision models. For each demonstration, we first segment the objects of interactions in the initial frame using a language prompt with Grounded-SAM [47]. The language prompts used for each task are described in Section IV-A. Next, we track the segmented objects across frames using CoTracker [48], which produces 2D object points for each demonstration. Finally, given per-frame depth, we unproject these 2D points into 3D, effectively obtaining 3D point object point clouds across time. While this process is straightforward for in-scene demonstrations, the Aria glasses do not provide depth. Therefore, for in-the-wild demonstrations, we use FoundationStereo [49], a framework that estimates a disparity map from rectified stereo images and the baseline between the cameras. For this, we use streams from the two front-facing SLAM cameras, rectify them, and use the translation norm provided by the Aria glasses as the baseline B. These inputs are passed to FoundationStereo to obtain a disparity map d with respect to the left SLAM camera. Using classical stereo geometry, the depth relative to the left frame, Z, is then recovered as: $Z = (f \cdot B)/d$, where f is the focal length of the left camera. 2D object tracks can then be unprojected to this estimated depth for in-the-wild demonstrations. For consistency, we transform all in-scene points into the robot base frame and all in-the-wild demonstrations into the world frame assigned during data collection. This ensures that all points lie on a similar horizontal plane since Aria glasses use the gravity vector to assign the world frame (explained in Section III-B.1).

3) Domain Alignment: 3D object tracks calculated with respect to the Aria glasses vary across demonstrations in the in-the-wild dataset, particularly due to differences in the height of the manipulated objects or the user collecting the data. To address this issue, we transform all 3D points into the robot base frame before training policies, using the in-scene demonstrations as an anchor.

Each demonstration consists of a trajectory of object points $\mathcal{O}^t \in \mathbb{R}^{N\times 3}$ and fingertip points $\mathcal{F}^t \in \mathbb{R}^{5\times 3}$, where N is the number of objects, fixed at 500 across all tasks. We refer to in-the-wild trajectories as $\mathcal{T}_w = \{\mathcal{O}_w^t, \mathcal{F}_w^t\}$ and in-scene trajectories as $\mathcal{T}_s = \{\mathcal{O}_s^t, \mathcal{F}_s^t\}$. To transform these trajectories into a uniform space, given a single in-scene trajectory $\mathcal{T}_s$ and an in-the-wild trajectory $\mathcal{T}_w$, we compute the translation between the centroids of the object points in their first frames, $\Delta\mathcal{O} = \mathcal{O}_s^0 - \mathcal{O}_w^0$. We then translate the in-the-wild trajectory by this offset, yielding $\hat{\mathcal{T}}_w = \{\mathcal{O}^t + \Delta\mathcal{O}, \mathcal{F}^t + \Delta\mathcal{O}\}$. This aligns the centroids of the object point clouds. However, since the world frame's rotation around gravity is assigned arbitrarily, relying solely on this translation can lead to large variations in z-axis orientation. This may cause demonstrations where the initial hand pose is fully rotated and object positions appear swapped. Figure illustrating this issue can be found on our website.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

To estimate a reliable rotation around the z-axis, we use the initial hand poses of both trajectories, $\mathcal{F}_s^0$ and $\mathcal{F}_w^0$, and apply the Kabsch algorithm [50] to compute the rigid transform between them. From this transform, we extract the rotation around the z-axis R_z, and apply it to the in-the-wild demonstrations, yielding the final transformed trajectories:

$$\hat{\mathcal{O}}_w^t = R_z \cdot \mathcal{O}_w^t + \Delta\mathcal{O} \tag{1}$$

$$\hat{\mathcal{F}}_w^t = R_z \cdot \mathcal{F}_w^t + \Delta\mathcal{O} \tag{2}$$

$$\hat{T}_w = \{\hat{\mathcal{O}}_w^t, \hat{\mathcal{F}}_w^t\} \tag{3}$$

We apply this transformation to every in-the-wild demonstration, and both in-the-wild and in-scene demonstrations are then used for policy learning, as described in the next section.

C. Learning and Deploying Smart Glass Policies on Robots

1) Policy Learning: To handle visual differences between the robot environment and in the wild human demonstrations, AINA utilizes transformer-based point-cloud policies and builds on top of the state-of-the-art imitation learning algorithm Point-Policy [7]. We provide the policy with a trajectory of fingertips $\mathcal{F}^{t-T_o:t}$ and object points $\mathcal{O}^{t-T_o:t}$ as input, and train the model to predict the subsequent fingertip trajectory $\mathcal{F}^{t:t+T_p}$, where $T_o = 10$ and $T_p = 30$ denote the observation history and prediction horizon, respectively. In our architecture, the observation history for each point is encoded into a single vector using Vector Neuron Multilayer Perceptrons (MLPs) [51]. These differ from regular MLPs in two key ways: (1) points are represented with 3D perceptrons rather than 1D, and (2) they employ SO(3)-equivariant activation layers. We choose vector neuron MLPs due to their demonstrated ability to better capture 3D geometric information [51]. The flattened vectors are then passed into a transformer encoder as tokens. Positional encoding is learned for only fingertip tokens and not keypoint tokens. The representations output by this encoder are subsequently fed into an MLP to predict the future fingertip trajectory. Mathematically, this can be expressed as follows:

$$\hat{\mathcal{F}}^{t:t+T_p} = \pi\big(\mathcal{F}^{t-T_o:t}, \mathcal{O}^{t-T_o:t}\big). \tag{4}$$

The entire system is trained end-to-end in a supervised manner using the mean squared error between the predicted and the ground-truth fingertips:

$$\mathcal{L}_{\text{MSE}} = \mathbb{E}\left[\left(\mathcal{F}^{t:t+T_p} - \hat{\mathcal{F}}^{t:t+T_p}\right)^2\right]. \tag{5}$$

In order to improve generalization, we apply augmentations during training. For each datapoint, we uniformly sample a 3D translation in the range $[-30\text{cm}, 30\text{cm}]$, a scaling factor in the range $[0.8, 1.2]$, and a rotation between $[-60°, 60°]$ around the gravity axis. These augmentations are combined into a single transformation, which is then applied consistently to both the input to the model and the ground truth output used to calculate L_{MSE}. Finally, to prevent the model from overfitting to the fingertips, we add Gaussian noise in the range $[-2\text{cm}, 2\text{cm}]$ to the input fingertips, but not to the predicted actions. We train the model for 2000 epochs, which typically takes about 2 hours per task.

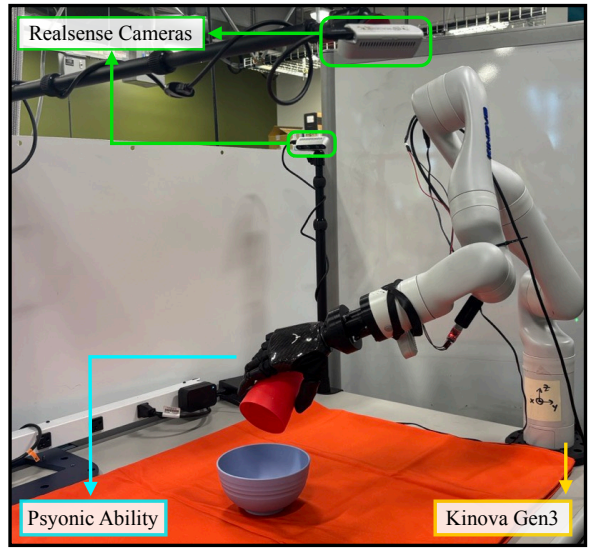

Fig. 5: Illustration of our robot setup.

2) Human Policy → Robot Deployment:

*a) **Robot Setup**:* Our robot setup consists of a single Kinova Gen3 robot arm [52] with 7 degrees of freedom (DOF) and a Psyonic Ability Hand with five fingers [3]. The Ability Hand has six DOFs: one in each finger and two in the thumb. It is designed as a prosthetic hand, making it compact and similar in size to a human hand. To observe the robot's environment, we use two RealSense RGB-D cameras placed around the operation space. Our robot configuration is illustrated in Fig. 5.

*b) **Inverse Kinematics**:* The kinematics of human arms and hands differ from those of tabletop manipulators and robot hands, making it non-trivial to replay human trajectories on a robot. Although the Ability Hand's small size reduces this embodiment gap, the lack of wrist joints in tabletop manipulators means that naively moving the arm and hand separately often leads to infeasible configurations. To address this, we implemented a custom full arm–hand inverse kinematics (IK) module $\mathcal{I}$, similar to [2]. Given desired fingertips $\mathcal{F}^{t+1} \in \mathbb{R}^{5\times3}$ and current Kinova and Ability joints $\mathcal{J}^t \in \mathbb{R}^{13}$, the module outputs next joint angles $\mathcal{J}^{t+1} = \mathcal{I}(\mathcal{F}^{t+1}, \mathcal{J}^t)$. The policy predicts fingertips as actions, and the resulting joint angles are applied to the robot during deployment. As in training, we segment and track objects in 3D to obtain object points and use forward kinematics to compute the fingertips.

*c) **Practical Implementation Details**:* Since the human demonstrations do not include force information, for tasks involving grasping, we set a grasping threshold: if the distance between the thumb and any other finger is less than 5 cm, the fingers are moved closer together. This helps mimic the force that humans apply during grasps.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

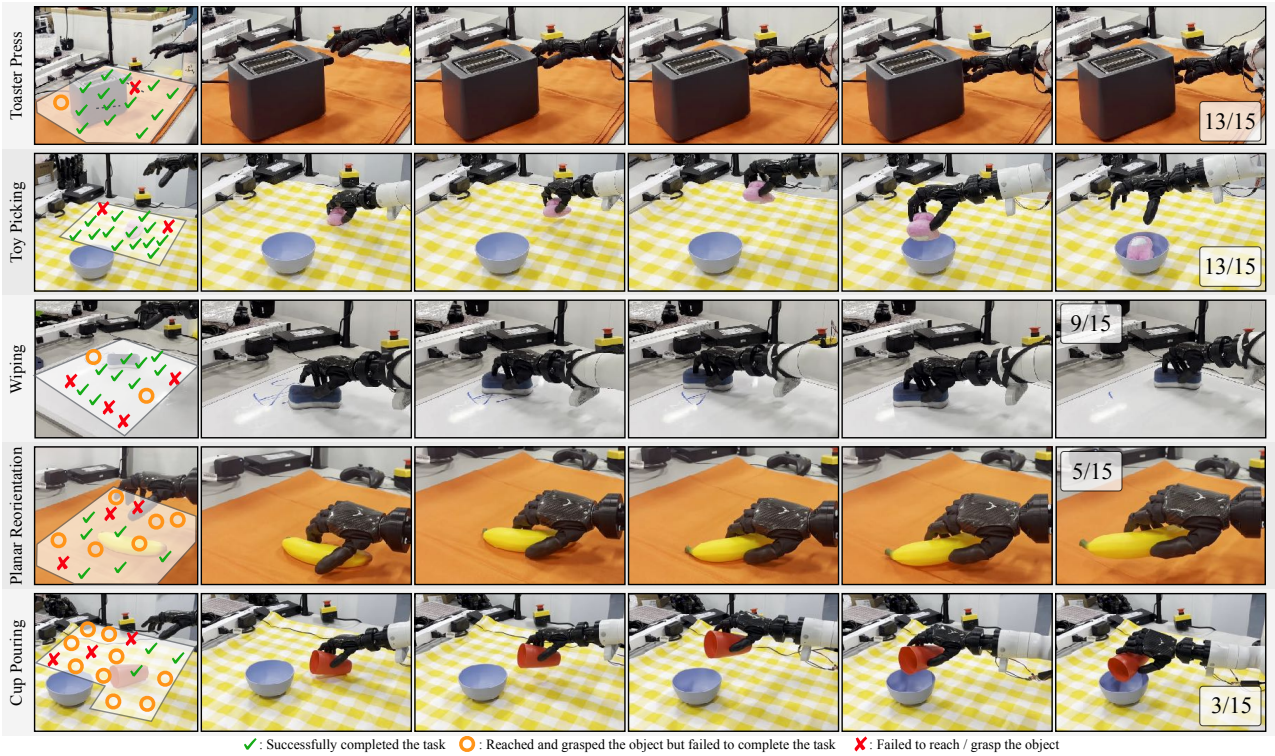

Fig. 6: Robot rollouts of AINA across five tasks. Spatial generalization is shown in the leftmost column for each task. The meaning of each symbol is explained below the figure. For the Toaster Press task, dotted lines indicate the toaster's orientation; when not shown, the orientation remains the same as in the showcased rollout. Rollouts of more tasks can be found on https://aina-robot.github.io.

IV. Experimental Evaluation

We perform various real robot experiments and compare AINA against multiple baselines to answer the following questions:

1) How important are the different types of data used in AINA?
2) How well does AINA perform when the height of the operation space changes?
3) How does AINA compare to image-based approaches for learning from human data?

A. Task Descriptions

We evaluate AINA on total of nine tasks, each chosen to represent a distinct skill or motion modality (wiping, pick-place, reorientation) and to reflect common daily manipulation activities. Robot rollouts, success rates, and spatial generalization results for five of our task are shown in Fig. 6. The rest of our tasks include *Oven Turning*, *Stowing*, *Drawer Opening*, and *Oven Opening*. More information about these tasks can be found on our website.

B. How important are the different types of data used in AINA?

AINA is a new framework for learning robot policies by co-training on in-the-wild and in-scene human video demonstrations. In-scene demonstrations are used both to standardize the input observations and to improve the policy. In this section, we evaluate the importance of this recipe.

We compare AINA against the following baselines and present the results in Table I:

1) **In-Scene Only** [2]: A policy trained using only a single in-scene demonstration. Unlike HuDOR [2], we do not apply any reinforcement learning for this baseline.
2) **In-The-Wild Only** [31]: A policy trained solely on in-the-wild demonstrations. These demonstrations are recorded with respect to the initial frame of the RGB camera and then transformed into the robot space by measuring the distance from the left camera to the center of the operation space and shifting the points accordingly. The closest approach to this is EgoZero [31], but our baseline differs in two key ways: (a) we do not use ArUco markers for data transfer, and (b) we perform closed-loop tracking of all the object points.
3) **In-Scene Transform and In-The-Wild** [11]: A policy that does not use in-scene data during training, but uses the in-scene demonstration for transforming the in-the-wild demonstrations. This baseline is inspired by ZeroMimic [11] that trains policies with in-the-wild human videos and uses a single in-scene goal image to condition the framework.
4) **In-Scene Training and In-The-Wild**: A policy that does not use in-scene data for transformation but includes it during training. The transformation is done as described for the *Without In-Scene* baseline.

From these results, we make the following observations:

In-the-wild demonstrations improve spatial generalization. The *In-Scene Only* baseline succeeds when objects are placed

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

TABLE I: Comparison of success of AINA to policies trained with different datasets. All methods are evaluated in similar deployment scenarios, with minimum of 10 trials each.

Tasks	Toaster Press	Toy Picking
In-Scene Only [2]	30%	10%
In-The-Wild Only [31]	0%	0%
In-Scene Transform and In-The-Wild [11]	0%	10%
In-Scene Training and In-The-Wild	60%	20%
In-Scene and In-The-Wild (AINA)	**86%**	**86%**

close to the demonstrated position, but it fails to generalize beyond that location.

In-scene demonstrations improve training. Since deployment is performed using RGB-D cameras rather than Aria glasses, the actions predicted by the *In-The-Wild Only* and *In-Scene Transform* baselines appear highly misaligned, leading to behaviors that look out of distribution.

In-scene demonstrations help transform in-the-wild demonstrations. The in-the-wild data used here is collected on different surfaces, with varying heights and different initial head frames. This makes the transformation in the *In-The-Wild Only* baseline prone to unstable rotations of the object points, resulting in less accurate policies.

C. How does AINA compare to image-based approaches for learning from human data?

AINA uses object-centric point clouds as input to reduce the visual disparity between human and robot observations. To evaluate the impact of using point clouds, we compare AINA to two image-based (with or without history) architectures (BAKU [53]) on two of our tasks, with the results shown in Table II.

TABLE II: Comparison of success of AINA to policies trained with RGB images as input.

Tasks	Oven Opening	Drawer Opening
Masked BAKU	6/15	1/15
Masked BAKU with History	0/15	0/15
AINA	**12/15**	**11/15**

We observe that AINA outperforms these image-based baselines on both tasks. Within the in-the-wild demonstrations, the human head naturally moves, whereas the robot's camera remains fixed during deployment. This discrepancy causes the Masked BAKU with History inputs to fall out of distribution relative to the training data, causing the policies to perform extremely poorly. Masked BAKU performs better, succeeding in nearly half of the trials; however, we still observe that viewpoint disparity between human demonstrations and robot deployment negatively affects performance. These demonstrate the importance of ingesting 3D inputs instead of images, for effective human-to-robot transfer.

D. How does AINA perform when the height of the operation space changes?

AINA does not assume prior knowledge about the height of the manipulated object, the data collector, or the robot's operation space. To demonstrate its use in operation spaces with different heights, we placed 3.5 cm tall plates on top of the robot's desk to create three height levels, as illustrated in Fig. 7. For each height level, we collect an additional in-scene human demonstration for alignment (requiring less than a minute to collect), as described in Section III-B.1 and use the same human data originally collected in-the-wild. We show the results in Table III.

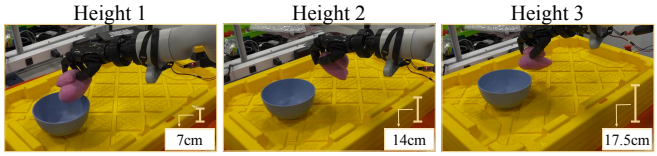

Fig. 7: Illustration of the height experiments. Each yellow plate is 3.5 cm tall. *Height 1* consists of 2 plates, *Height 2* of 4 plates, and *Height 3* of 5 plates. Thus, Height 1 is closest to the original deployment scenario, while Height 3 is the furthest.

TABLE III: Success rate of AINA deployed on plates with different height levels.

Tasks	Toy Picking	Wiping
Height 1	5/10	5/10
Height 2	6/10	5/10
Height 3	2/10	8/10

We find that the resulting policies perform robustly across tasks, reliably generalizing across heights. This demonstrates the flexibility of AINA in transferring manipulation skills from in-the-wild data to new scenarios with minimal human effort. Occasional failures arise when an in-scene human demonstration trajectory diverges significantly from the distribution of in-the-wild data. For example, in the Toy Picking task at Height 3, the in-scene demonstration brought the toy unusually close to the bowl. This atypical trajectory led the policy to reproduce the behavior during deployment, causing the toy to push the bowl.

E. More experimental results on object generalization

We evaluated the generalization of AINA policies by deploying existing policies zero-shot in environments with new objects. Due to space constraints, we provide the results of this experiment on our website: `https://aina-robot.github.io`.

V. Discussion, Limitations, Conclusion

In this work, we presented AINA, a new framework that leverages capabilities of Aria Gen 2 glasses to learn point-based multi-fingered policies from explicitly in-the-wild human demonstrations.

While promising, we observe three limitations. First, our framework cannot easily integrate force feedback, since hand pose estimation alone cannot capture this information, which is often crucial for accurate dexterous manipulation [54]–[56]. Second, the Aria Gen 2 glasses exhibit a slight difference in shutter timing between the RGB and SLAM cameras. Rapid head movements during data collection can therefore cause misalignment between the object's pixels in the RGB image and the corresponding depth in SLAM. To mitigate

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.

this, we currently instruct data collectors to avoid rapid head movements, though alternative solutions include using more robust 3D object tracking algorithms [57]. Finally, during deployment we currently use Realsense cameras due to difficulty of obtaining real-time depth estimates with FoundationStereo. However, this is an ongoing effort and we believe that with sufficient optimizations, we can receive near real-time depth.

REFERENCES

[1] C. Wang, *et al.*, "Dexcap: Scalable and portable mocap data collection system for dexterous manipulation," in *RSS*, 2024.

[2] I. Guzey, *et al.*, "Bridging the human to robot dexterity gap through object-oriented rewards," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 3344–3351.

[3] "Psyonic ability hand," 2023. [Online]. Available: https://www.psyonic.io/ability-hand

[4] A. Zorin, *et al.*, "Ruka: Rethinking the design of humanoid hands with learning," *Robotics: Science and Systems (RSS)*, 2025.

[5] C. Wang, *et al.*, "Mimicplay: Long-horizon imitation learning by watching human play," *arXiv preprint arXiv:2302.12422*, 2023.

[6] S. Park, *et al.*, "Demodiffusion: One-shot human imitation using pre-trained diffusion policy," 2025. [Online]. Available: https://arxiv.org/abs/2506.20668

[7] S. Haldar *et al.*, "Point policy: Unifying observations and actions with key points for robot manipulation," *arXiv preprint arXiv:2502.20391*, 2025.

[8] H. Bharadhwaj, *et al.*, "Towards generalizable zero-shot manipulation via translating human interaction plans," 2023.

[9] S. Bahl, *et al.*, "Affordances from human videos as a versatile representation for robotics," in *CVPR*, 2023.

[10] H. Bharadhwaj, *et al.*, "Track2act: Predicting point tracks from internet videos enables generalizable robot manipulation," in *ECCV*, 2024.

[11] J. Shi, *et al.*, "Zeromimic: Distilling robotic manipulation skills from web videos," in *International Conference on Robotics and Automation (ICRA)*, 2025.

[12] Y. Li, *et al.*, "Nimble: a non-rigid hand model with bones and muscles," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 4, pp. 1–16, 2022.

[13] S. P. Arunachalam, *et al.*, "Holo-dex: Teaching dexterity with immersive mixed reality," in *ICRA*, 2023.

[14] A. Iyer, *et al.*, "Open teach: A versatile teleoperation system for robotic manipulation," in *CoRL*, 2024.

[15] K. He, *et al.*, "Masked autoencoders are scalable vision learners," in *CVPR*, 2022.

[16] H. Wu, *et al.*, "Unleashing large-scale video generative pre-training for visual robot manipulation," 2023.

[17] N. Karaev, *et al.*, "Cotracker: It is better to track together," in *ECCV*, 2024.

[18] C. Doersch, *et al.*, "Tap-vid: A benchmark for tracking any point in a video," in *NeurIPS*, 2022.

[19] Y. Ye, *et al.*, "What's in your hands? 3d reconstruction of generic objects in hands," in *CVPR*, 2022.

[20] G. Pavlakos, *et al.*, "Reconstructing hands in 3d with transformers," in *CVPR*, 2024.

[21] K. Grauman, *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," in *CVPR*, 2022.

[22] P. Banerjee, *et al.*, "HOT3D: Hand and object tracking in 3D from egocentric multi-view videos," *CVPR*, 2025.

[23] K. Grauman, *et al.*, "Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives," in *CVPR*, 2024.

[24] S. Nair, *et al.*, "R3m: A universal visual representation for robot manipulation," in *CoRL*, 2022.

[25] T. Xiao, *et al.*, "Masked visual pre-training for motor control," *arXiv preprint arXiv:2203.06173*, 2022.

[26] S. Karamcheti, *et al.*, "Language-driven representation learning for robotics," in *RSS*, 2023.

[27] K. Shaw, *et al.*, "Videodex: Learning dexterity from internet videos," in *CoRL*, 2023.

[28] M. K. Srirama, *et al.*, "Hrp: Human affordances for robotic pre-training," in *RSS*, 2024.

[29] J. Engel, *et al.*, "Project aria: A new tool for egocentric multi-modal ai research," *arXiv preprint arXiv:2308.13561*, 2023.

[30] S. Kareer, *et al.*, "Egomimic: Scaling imitation learning via egocentric video," 2024. [Online]. Available: https://arxiv.org/abs/2410.24221

[31] V. Liu, *et al.*, "Egozero: Robot learning from smart glasses," 2025. [Online]. Available: https://arxiv.org/abs/2505.20290

[32] "Aria gen 2 glasses," 2025. [Online]. Available: https://ai.meta.com/blog/aria-gen-2-research-glasses-under-the-hood-reality-labs/

[33] K. Shaw, *et al.*, "Leap hand: Low-cost, efficient, and anthropomorphic hand for robot learning," *arXiv preprint arXiv:2309.06440*, 2023.

[34] I. Akkaya, *et al.*, "Solving rubik's cube with a robot hand," *arXiv preprint arXiv:1910.07113*, 2019.

[35] S. Yang, *et al.*, "Ace: A cross-platform visual-exoskeletons system for low-cost dexterous teleoperation," in *CoRL*, 2024.

[36] I. Guzey, *et al.*, "See to touch: Learning tactile dexterity through visual incentives," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 13 825–13 832.

[37] S. Chen, *et al.*, "Arcap: Collecting high-quality human demonstrations for robot learning with augmented reality feedback," *arXiv preprint arXiv:2410.08464*, 2024.

[38] T. Tao, *et al.*, "Dexwild: Dexterous human interactions for in-the-wild robot policies," *Robotics: Science and Systems (RSS)*, 2025.

[39] A. Brohan, *et al.*, "Rt-1: Robotics transformer for real-world control at scale," in *RSS*, 2023.

[40] Z. Mandi, *et al.*, "Cacti: A framework for scalable multi-task multi-scene visual imitation learning," *arXiv preprint arXiv:2212.05711*, 2022.

[41] M. Shridhar, *et al.*, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 785–799.

[42] Y. Ze, *et al.*, "Gnfactor: Multi-task real robot learning with generalizable neural feature fields," in *Conference on robot learning*. PMLR, 2023, pp. 284–301.

[43] T. Gervet, *et al.*, "Act3d: 3d feature field transformers for multi-task robotic manipulation," *arXiv preprint arXiv:2306.17817*, 2023.

[44] M. Lepert, *et al.*, "Shadow: Leveraging segmentation masks for zero-shot cross-embodiment policy transfer," in *Conference on Robot Learning (CoRL)*, Munich, Germany, 2024.

[45] G. Pavlakos, *et al.*, "Reconstructing hands in 3d with transformers," in *CVPR*, 2024.

[46] J. Romero, *et al.*, "Embodied hands: Modeling and capturing hands and bodies together," *arXiv preprint arXiv:2201.02610*, 2022.

[47] T. Ren, *et al.*, "Grounded sam: Assembling open-world models for diverse visual tasks," *arXiv preprint arXiv:2401.14159*, 2024.

[48] N. Karaev, *et al.*, "Cotracker3: Simpler and better point tracking by pseudo-labelling real videos," *arXiv preprint arXiv:2410.11831*, 2024.

[49] B. Wen, *et al.*, "Foundationstereo: Zero-shot stereo matching," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 5249–5260.

[50] J. Lawrence, *et al.*, "A purely algebraic justification of the kabsch-umeyama algorithm," *Journal of research of the National Institute of Standards and Technology*, vol. 124, p. 1, 2019.

[51] C. Deng, *et al.*, "Vector neurons: A general framework for so (3)-equivariant networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 200–12 209.

[52] "Kinova gen 3," 2010. [Online]. Available: https://www.kinovarobotics.com/product/gen3-robots

[53] S. Haldar, *et al.*, "Baku: An efficient transformer for multi-task policy learning," *arXiv preprint arXiv:2406.07539*, 2024.

[54] C. Higuera, *et al.*, "Tactile beyond pixels: Multisensory touch representations for robot manipulation," *arXiv preprint arXiv:2506.14754*, 2025.

[55] A. Sharma, *et al.*, "Self-supervised perception for tactile skin covered dexterous hands," *arXiv preprint arXiv:2505.11420*, 2025.

[56] I. Guzey, *et al.*, "Dexterity from touch: Self-supervised pre-training of tactile representations with robotic play," *arXiv preprint arXiv:2303.12076*, 2023.

[57] L. Jin, *et al.*, "Stereo4d: Learning how things move in 3d from internet stereo videos," *arXiv preprint*, 2024.

Authorized licensed use limited to: ShanghaiTech University. Downloaded on October 19,2026 at 15:57:21 UTC from IEEE Xplore. Restrictions apply.