Learning to Generalize via Human Manipulation Priors

Mohan Kumar Srirama

THE ROBOTICS INSTITUTE
SCHOOL OF COMPUTER SCIENCE
CARNEGIE MELLON UNIVERSITY
Pittsburgh, PA

Thesis Committee:

Prof. Deepak Pathak (Chair)
Prof. Abhinav Gupta
Mihir Prabhudesai

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Robotics.

Abstract

Generalization remains a core challenge in robotics: enabling robots to adapt to new objects, environments, and embodiments with minimal additional data. This thesis explores how human prior knowledge, captured through both passive observation and active demonstration, can be used to address this challenge. We propose two complementary approaches that scale robot learning using large-scale human-derived data.

First, we present **HRP: Human Affordances for Robotic Pre-Training**, a method that learns visual affordances from internet-scale human videos. By automatically extracting hand trajectories, contact points, and object labels, we pre-train a vision transformer with structured human priors. When fine-tuned for robotic control, these representations yield over 15% absolute gains in real-world tasks and generalize effectively across camera viewpoints and robot morphologies.

Second, we introduce **DexWild: Dexterous Human Interactions for In-the-Wild Robot Policies**, which scales robot learning through in-situ human demonstrations. Using a lightweight wearable device (DexWild-System), we collect diverse, high-fidelity demonstrations of natural tasks. A co-training algorithm combines this human data with a smaller set of robot-specific examples, enabling generalization to new scenes, object categories, and robot hands. DexWild-trained policies achieve 68.5% success on unseen tasks, nearly four times higher than robot-only training, and show a 5.8 times improvement in cross-embodiment generalization.

Together, these results show that human priors, whether learned passively from video or actively through demonstration, significantly enhance a robot's ability to generalize beyond its training domain. We conclude by outlining directions for combining these methods and incorporating richer sensory inputs.

Acknowledgments

I would first like to express my deepest gratitude to my incredible advisors, Professor Deepak Pathak and Professor Abhinav Gupta, for their guidance, support, and belief in me throughout my time at Carnegie Mellon University. Their mentorship has been instrumental in shaping my thinking, research philosophy, and academic journey. I have learned immensely from their insight, rigor, and vision, and this thesis would not have been possible without their encouragement and trust.

I am also sincerely grateful to the senior PhD students who mentored me along the way: Sudeep Dasari, Shikhar Bahl, and Kenny Shaw. Their generosity in sharing knowledge, their honest feedback, and their patience during my steep learning curves made a lasting impact on my growth as a researcher. I have looked up to each of them, and their guidance has helped me through some of the toughest and most rewarding phases of this work.

To all my fellow co-authors and labmates: Tony Tao, Yulong Li, Jim Yang, Jason Liu, Sasha Khazatsky, Max Sobol Mark, Russel Mendonca, Gaoyue Zhou, Jianren Wang, Victoria Dean, Soroush Nasiriany, Jared Mejia, Lili Chen, Mihir Prabhudesai, Gokul Swamy, Mrinal Verghese, Ananye Agrawal, Justin Wasserman, Judy Ye, Homanga Bharadwaj, Murtaza Dalal, Jayesh Singla, Himangi Mittal, Chen Bao, Haoyu Xiong, Hengkai Pan, Sandeep Routray, Raunaq Bhirangi, Adam Kan, and Abby Defranco, thank you for the camaraderie, collaboration, and stimulating discussions that made this journey intellectually rich and emotionally rewarding. I feel lucky to have been surrounded by such passionate and brilliant peers.

I want to thank my roommates Madhusudhan Reddy Pittu, Parth Malpathak, and Siddharth Mehta for all the fun, late-night conversations, and support over the years. A special shoutout to Madhu, my partner in crime and constant source of encouragement, for making Pittsburgh feel like home.

Finally, I extend my heartfelt appreciation to my family, especially my parents Srirama and Malini, and my sister Shalini, whose unwavering support, encouragement, and steady presence kept me grounded throughout this journey. Your belief in me has meant more than words can express.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Roboticists have long dreamed of creating robots that can perform tasks with the same dexterity and adaptability as humans. We would like robots to deftly generalize to many different objects, environments, and embodiments — yet this vision of truly versatile robot behaviors remains a formidable challenge. Current robot learning paradigms struggle to scale, primarily due to the difficulty of collecting and generalizing from sufficiently large and diverse datasets [5, 6]. In contrast, other fields like natural language processing and computer vision have achieved breakthroughs by harnessing vast datasets [7].

This thesis addresses the question: *How can we leverage human-derived data to bridge the data diversity gap and enable extreme generalization in robotics?*

Many recent efforts in robotics have focused on two broad strategies: leveraging large-scale *teleoperation* to collect robot data, and exploiting *human video* datasets for representation learning. Teleoperation can provide high-quality, on-policy demonstrations for robot learning [8], but it requires expert operators and specialized setups, making it labor-intensive and not easily scalable [9, 10]. Additionally, collecting robot data across diverse environments (each requiring moving hardware) is prohibitively slow and expensive [6].

On the other hand, internet-scale video collections of humans (e.g., [1], [11]) offer a rich source of prior knowledge. Prior works have attempted to train visual encoders on human videos and then transfer them to robots [12, 13]. While these approaches provide some benefits, recent analyses showed that straightforwardly applying self-supervised learning on human video often fails to outperform even ImageNet pre-trained features for downstream robotics [14, 15]. We hypothesize that we need to inject more *structured human knowledge* into the learning process in order to unlock generalization.

In this thesis, we explore two complementary approaches to inject human priors into robot learning:

(1) Learning from Human Videos (Affordance-Based Pre-Training): In Part I of this thesis, we mine large-scale *unlabeled human videos* to extract meaningful affordance cues that can guide robot perception. We develop a method to automatically annotate human videos with **affordance labels** — such as contact points between hands and objects, the pose of human hands during manipulation, and the identity and location of actively used objects. These labels, obtained with off-the-shelf vision models applied to massive video datasets, serve as distilled human prior knowledge about *how humans interact with their environment*. We then use these labels to **pre-train a visual representation** (specifically, a ViT-based encoder) via auxiliary tasks: the model is trained to predict the future hand trajectory, contact, and object usage from a single video frame. This approach, called **HRP: Human Affordances for Robotic Pre-Training** [16], injects an *inductive bias* for understanding hand-object interactions into the vision network. Part I (Chapter 2) demonstrates that fine-tuning any baseline vision model with these human affordance losses yields representations that dramatically improve downstream robot policy learning. We show across 5 real-world manipulation tasks (including block stacking, pouring, etc., using three different robot embodiments) that adding HRP pre-training boosts success rates by a minimum of 15% and up to 25%, even in out-of-distribution test scenarios [**? ?**]. Notably, these gains hold across multiple camera views (e.g., egocentric and third-person) [**?**], indicating a robust improvement in learned visual representations.

(2) Learning from In-the-Wild Human Demonstrations (Motion Capture System): In Part II, we focus on directly leveraging *human demonstrations* to teach robots, bypassing the need for expensive teleoperation. We introduce **DexWild**, a framework in which human operators perform everyday manipulation tasks naturally, using their own hands, while wearing a lightweight motion-capture system (DexWild-System). This system records high-fidelity 3D hand movements and interactions at scale [**? ?**]. We collect a large dataset of human multi-modal demonstrations across many homes and settings, capturing a diversity of objects and action strategies far beyond typical robot-collected data. To transfer these human skills to robots, we propose a **co-training approach** where a policy is trained on both the human demonstration dataset and a smaller set of matched robot demonstrations (to "ground" the policy in the robot's embodiment) [**? ?**]. The resulting policies, trained with our DexWild framework, achieve striking generalization: in experiments, DexWild-trained policies attain a 68.5% success rate on completely unseen environment-task combinations — nearly four times higher than policies trained on robot data alone [**?**]. They also generalize across embodiments (e.g., a policy trained with one robot hand works on a different hand with minimal adaptation), showing 5.8× better cross-embodiment performance than baselines [**?**]. Part II (Chapter 3) details how combining human and robot data in training leads to robust, versatile manipulation skills that outperform state-of-the-art single-dataset policies.

In summary, this thesis shows that **human priors can be a powerful catalyst for robot gen-**

eralization. By pre-training on human video affordances and by co-training with human demonstrations, we infuse robot learning with knowledge of how humans perceive and act on the world. This yields robot policies that require less robot-specific data and yet excel in novel scenarios. In the final chapter, we will discuss how these approaches complement each other and outline future directions — such as integrating affordance-based pre-training with demonstration learning, incorporating human corrective feedback, and extending to multi-agent settings — that could further exploit the rich well of human experience for robotic generalization.

Part I

Human Affordances for Robotic Pre-Training (HRP)

Chapter 2

HRP: Human Affordances for Robotic Pre-Training

2.1 Introduction

A truly generalist robotic agent must acquire diverse manipulation skills (ranging from block stacking to pouring) that work with novel objects and remain robust to realistic environmental disturbances (e.g., lighting changes, small camera shifts). Due to the scale of this challenge, the field has trended towards learning these agents directly from data [20, 21], particularly robot trajectories collected either by expert demonstrators or autonomously by the agents themselves (via Reinforcement Learning [22]). Unfortunately, there are innumerable objects/environments, so roboticists cannot tractably collect enough real-world demonstration data and/or design a simulator that captures all this diversity.

One promising solution for this "data challenge" is for the robot to learn a *suitable representation* from Out-Of-Domain (OOD) data that can be transferred into the robotics domain. For example, prior work [17, 18, 19] trained self-supervised image encoders on large scale datasets of human videos (e.g., Ego4D [1]), using standard reconstruction objectives and contrastive learning [23] objectives – e.g., Masked Auto-Encoders [24] (MAE) and Temporal Contrastive Networks [25] (TCN) respectively – developed by the broader learning community. After pre-training, these representations are used to initialize downstream imitation learning [26] algorithms. This formula is extremely flexible, and can substantially reduce the amount of robot data required for policy learning. However, the representations are often only effective when using specific camera views and robot setups. Furthermore, independent evaluations [14, 15] recently showed that these representations cannot improve (on average) over the most obvious baseline – a self-supervised ImageNet representation [24, 27]!

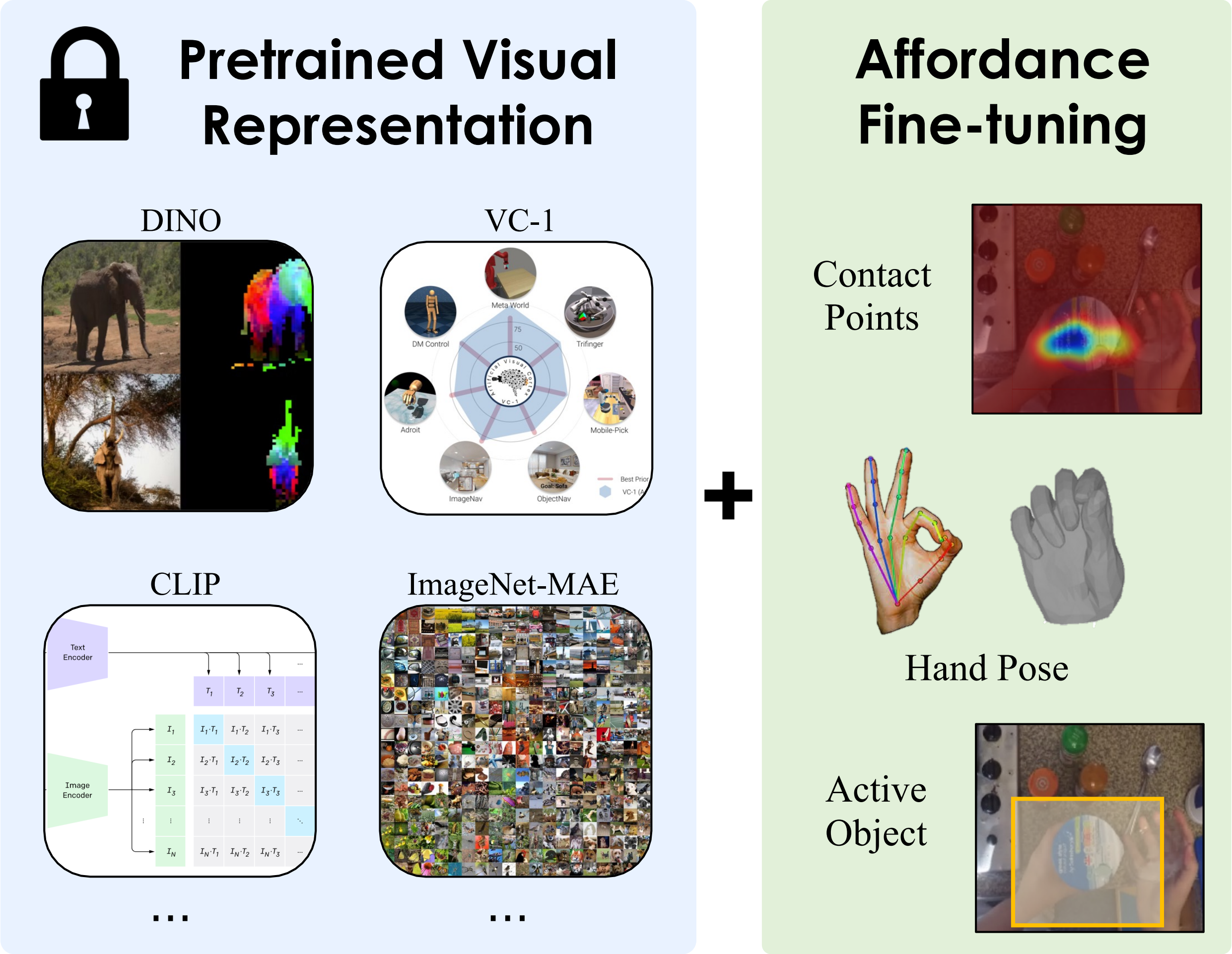

Figure 2.1: Pre-trained representations offer a scalable solution to the robotics data bottleneck [17, 18, 19], but existing methods fail to reliably improve over simple baselines like ImageNet [14, 15]. Thus, we present **HRP**, a method that mines affordances (e.g., contact, hand pose, and object labels) from human videos and uses them to improve self-supervised visual encoders. Our best HRP representation consistently outperforms 6 SOTA baselines by $\geq$ **20%** across 5 diverse tasks and 3 camera views.

This result is surprising since robot trajectories and human video sequences share so much common structure: both modalities contain an agent (e.g., human or robot) using their end-effector (e.g., human hand, robot gripper) to manipulate objects in their environment. Ideally, representations trained on this data would learn useful object attributes (e.g., where to grasp a mug), and spatial relationships between the end-effector and target objects. We hypothesize that traditional self-supervised learning objectives are unable to extract this information from human video data, and that explicitly predicting these object/spatial features would result in a stronger robotic representation (i.e., higher down-stream control performance). Our key insight is that abandoning self-supervision comes at minimal cost – the necessary object and hand labels can be scalably mined using off-the-shelf vision pipelines.

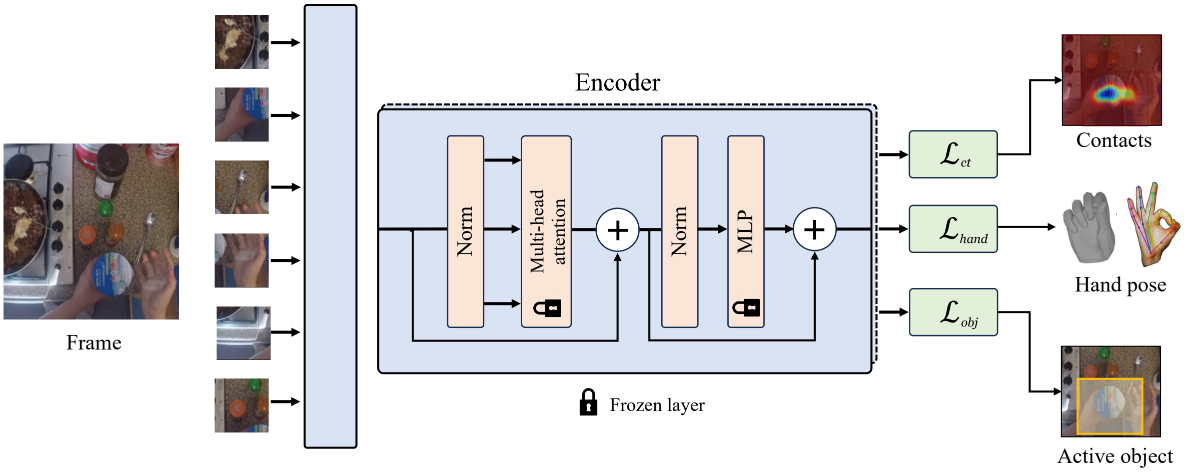

Figure 2.2: HRP fine-tunes a pre-trained encoder to predict three classes of human affordance labels via L2 regression. Specifically, the network must predict future contact points, human hand poses, and the target object given an input frame from the video stream. These affordance labels are mined autonomously from a human video dataset [1] using off-the-shelf vision detectors [2]. HRP representations are then fine-tuned to solve downstream manipulation tasks via behavior cloning.

This paper proposes Human affordances for Robotic Pre-training (HRP), a semi-supervised pipeline to learn effective robotic representations from human video. HRP works in two stages: first, it extracts hand-object "affordance" information – i.e., which objects in the scene are graspable and how the robot should approach them – from human videos using off-the-shelf tracking models [2, 28]. These affordances are then distilled into a pre-existing representation network (e.g., ImageNet MAE [24]), *before* the policy fine-tuning stage. This paradigm allows us to inject useful information into the vision encoder, while preserving the flexibility of self-supervised pre-training – i.e., all labels are automatically generated and the network can be easily slotted into downstream robotic policies/controllers via fine-tuning. To summarize, **we learn stronger robotic representations by predicting object interactions and hand motion from human video dataset images** (see Fig. 2.1). Our investigations and experiments lead to the following contributions:

1. We present a semi-supervised learning algorithm – HRP– that leverages off-the-shelf human affordance models to learn effective robotic representations from human video. The proposed pipeline strongly outperforms representations learned purely via self-supervision.

2. Applying HRP to 6 pre-existing representations (including ImageNet [27, 24], VC-1 [19], and DINO [29]) substantially boosts robot performance. This conclusion is backed by **3000+ robot trials**, and replicates across 3 camera views, 3 distinct robotic setups, and 5 manipulation tasks!

3. Our ablation study reveals that HRP's three affordance objectives (hand, object, and contact based loss terms) are all critical for effective representation learning.

4. We show that HRP representations generalize across different imitation learning stacks – HRP improves diffusion policy [3] performance by 20%!

5. Our best representation, which increases performance by 20% over State-of-the-Art (SOTA), will be fully open-sourced, along with all code and data.

2.2 Related Work

Representation Learning in Robotics End-to-end policy learning offers a scalable formula for acquiring robotic representations: instead of hand-designing object detectors or image features, a visual encoder is directly optimized to solve a downstream robotic task [20]. Numerous works applied this idea to diverse tasks including bin-picking [30, 31, 21], in-the-wild grasping [32, 33], insertion [5, 20], pick-place [34], and (non-manipulation tasks like) self-driving [35, 36, 37]. Furthermore, secondary learning objectives – e.g., dynamics modeling [38, 39], observation reconstruction [40], inverse modeling [41], etc. – can be easily added to improve data efficiency. While this paradigm can be effective, learning purely from robot data requires an expensive data collection effort (e.g., using an arm farm [31, 30], large-scale tele-operation [34], or multi-institution data collection [42, 43]), which is infeasible for (most) task settings.

To increase data efficiency, prior work applied self-supervised representation learning algorithms on out-of-domain datasets (like Ego4D [1]), and then fine-tuned the resulting representations to solve downstream tasks with a small amount of robot data – e.g., via behavior cloning on ≤ 50 expert demonstrations [17, 19, 18], directly using them as a cost/distance function to infer robot actions [44, 45], or directly pre-training robot policies from extracted human actions. [46, 47, 9]. While this transfer learning paradigm can certainly be effective, it is unclear if these robotic representations [19, 17, 18] provide a substantial boost over pre-existing vision baselines [14, 15], like ImageNet MAE [24] or DINO [29]. One potential issue is that roboticists often use the same exact pre-training methods from the vision community, but merely apply them to a different data mix (e.g., VC-1 [19] applies MAE [24] to Ego4D [1]). Thus, the resulting representations are never forced to key in on object/agent level information in the scene. This paper proposes a simple formula for injecting this information into a vision encoder, using a mix of hand and object affordance losses, which empirically boost performance on robotic tasks by 25%.

Affordances from Humans HRP is heavily inspired by the *affordance learning* literature in computer vision [48, 49]. These works use human data as a probe to learn environmental cues (i.e., affordances) that tell us how humans might interact with different objects. These include physical [50, 51, 52, 53, 54, 55, 56] and/or semantic [57, 58] scene properties, or forecast fu-

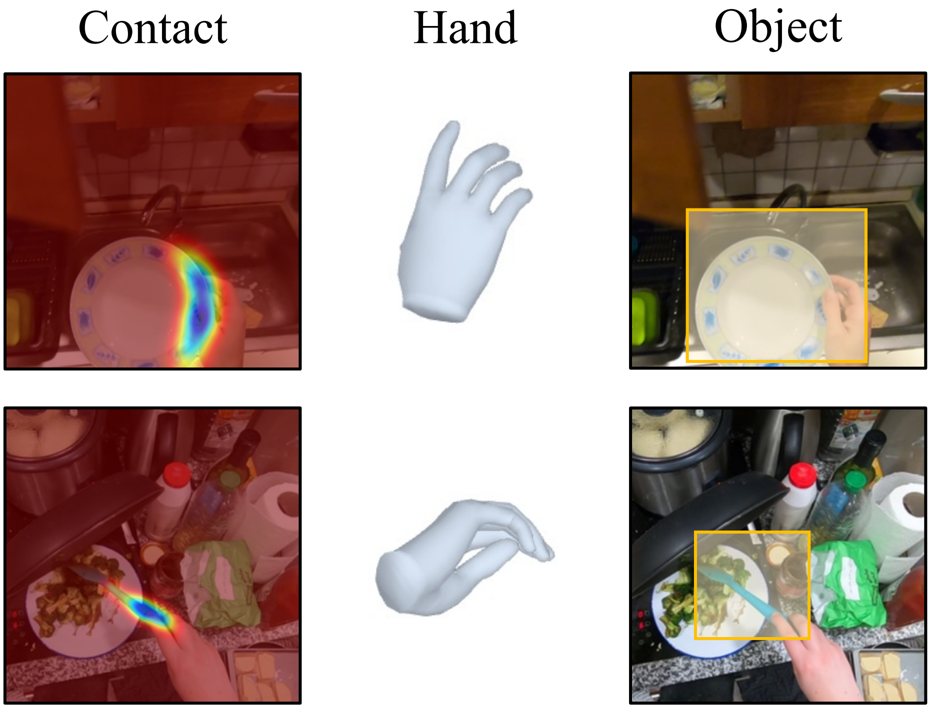

Figure 2.3: We extract 3 affordances – contact heatmaps, hand poses and active object bounding boxes – from human videos.

ture poses [59, 60, 61, 62, 63, 64, 65, 66, 67, 1, 68, 69, 70] Affordances can also be learned at object or part levels [71, 72, 73, 74, 75, 76]. Usually such approaches leverage human video datasets [1, 77, 78, 79] or use manually annotated interaction data [80, 81, 2]. In addition to these cues, robotic affordances must consider how to move before and after interaction [82, 83]. A simple, scalable way to capture this information is by detecting these cues from human hand poses in monocular video streams [84, 83, 28, 85], which show robots reaching for and manipulating diverse, target objects. Our method combines these three approaches to create a human affordance dataset automatically from human video streams. The labels generated during this process are distilled into a representation and used to improve downstream robotics task performance.

2.3 Preliminaries

2.3.1 Visual Representation Learning

Our goal is to learn a visual encoder network f_θ that takes an input image I and processes it into a low-dimensional vector $f_\theta(I) \in \mathcal{R}^d$. This resulting "embedding vector" would ideally encode important scene details for robotic policy learning – like the number and type of objects in a scene and their relationship to the robot end-effector. In this paper, f_θ is a transformer network (specifically ViT-B [86], with patch size 16 and $d = 768$) parameterized with network weights θ. But to

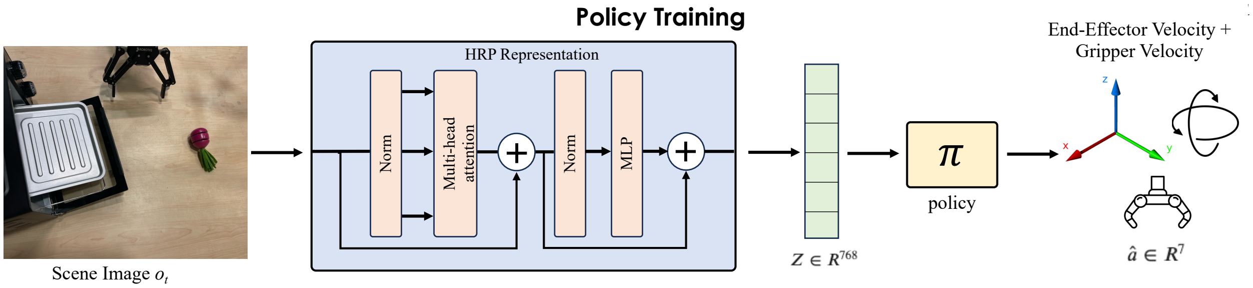

Figure 2.4: We present our policy training pipeline, which uses Behavior Cloning (BC) to train policy π, using optimal expert demonstrations. The image observation (o_t) is processed using our HRP representations resulting in a latent vector z. The policy uses z to predict end-effector velocity actions (delta ee-pose/gripper), which are directly executed on the robot during test-time.

be clear, all our methods are network architecture agnostic.

Self-Supervised Learning The computer vision community has broadly adopted *self-supervised* representation learning algorithms that can pre-train network weights without using *any* task-specific supervision. This can be accomplished using a *generative learning objective* [87], which trains f_θ alongside a decoder network D that reconstructs the original input image input from the representation. Another common approach is *contrastive learning* [23, 88], which optimizes f_θ to maximize the mutual information between the encoding and the input image (i.e., place "similar" images closer in embedding space). In practice, these methods can learn highly useful features for downstream vision tasks [24, 88], but struggle in robotics settings [14, 15]. Our goal is to inject these features into an existing self-supervised network, with an affordance-driven fine-tuning stage.

2.3.2 Extracting Affordance Labels from Human Data

Before we can do any fine-tuning, we must first curate a suitable human affordance dataset $\mathcal{D}_H$. Thankfully this task can be done automatically using off-the-shelf vision modules, applied to a set of $150K$ human-object interaction videos from Ego4D (originally sampled by **(author?)** [17]). These are subsets of larger videos (around 1.2K) videos, which were further broken down into shorter clips. Each clip contains a semantically meaningful action by the human. Each video clip V contains image frames $V = \{I_1, \ldots, I_T\}$ that depict human hands performing tasks and moving around in the scene. From these images, we obtain **contact locations**, **future hand p-oses**, and **active object labels** (examples in Fig. 2.3) that capture various agent-centric properties (how to move and interact) and environment centric properties (where to interact) at multiple scales, i.e. contact-level and object-level. The following sections detail how each of these labels were generated.

Contact Locations To extract contact locations for an image I_t (with no object contact), we find the frame $I_j; j > t$ where contact with a given object will begin, using a hand-object interaction detection model [2]. Then, we use I_j to find the active object O_j and the hand mask M_j. The points intersecting M_j and O_j (acquired via skin segmentation) are our contact affordances (C_j). To account for motion between I_t and I_j, we compute the homography matrix between the frames and project those points forward. This is done using standard SIFT feature tracking [89]: $C_t = H_{j,t}C_j$. *In other words, the contact locations denote where in I_t the human will contact in the future.* Note that there could be a different number of points for each contact scenario, which is non-ideal for learning. Thus, we fit a Gaussian Mixture Model with $k = 5$ modes on C_t to make a uniform contact descriptor – defined as the means c_t of the mixture model.

Future Hand Poses This affordance label captures how the human moves next (e.g., to complete a task or reach an object), as the video V progresses. Given a current frame I_t, we detect the human hand's 2d wrist position (h_{t+k}) in a future frame I_{t+k}, where usually $k = 30$ (empirically determined). This is done using the Frank Mocap [28] hand detector. To correctly account for the human's motion, these wrist points are back-projected (again using the camera homography matrix) to I_t to create the final "future wrist label," $h_t = H_{t+k,t}h_{t+k}$.

Active Object Labels In a similar manner to the contact location extraction, we run a hand-object interaction detection model [2] on V to find the image where contact began I_c. The same detector is used to find the four bounding box coordinates of the object that is being interacted with, which we refer to as the "active object." These coordinates b_c are then projected to every other frame I_t, using the homography matrix (see above). This results in an active bounding box b_t for each image in V.

2.4 Introducing HRP

A variety of visual pre-training tasks have been shown to help with downstream robotic performance–ranging from simple ImageNet classification [90] to self-supervised learning on human video [18, 17, 44, 19, 91]. Although these approaches operate on human videos and simple image frames, they fail to explicitly model the rich hand-object contacts depicted. In contrast, we believe explicitly modeling the *affordances* [49] in this data could allow us to learn useful information about the agent's intents, goals, and actions. Indeed, past work has shown that affordances can act as strong prior for manipulation [92, 93, 94, 95, 82, 96, 97, 98] in general. Moreover, this information can be represented in many different formats, such as physical attributes, geometric properties, inter-

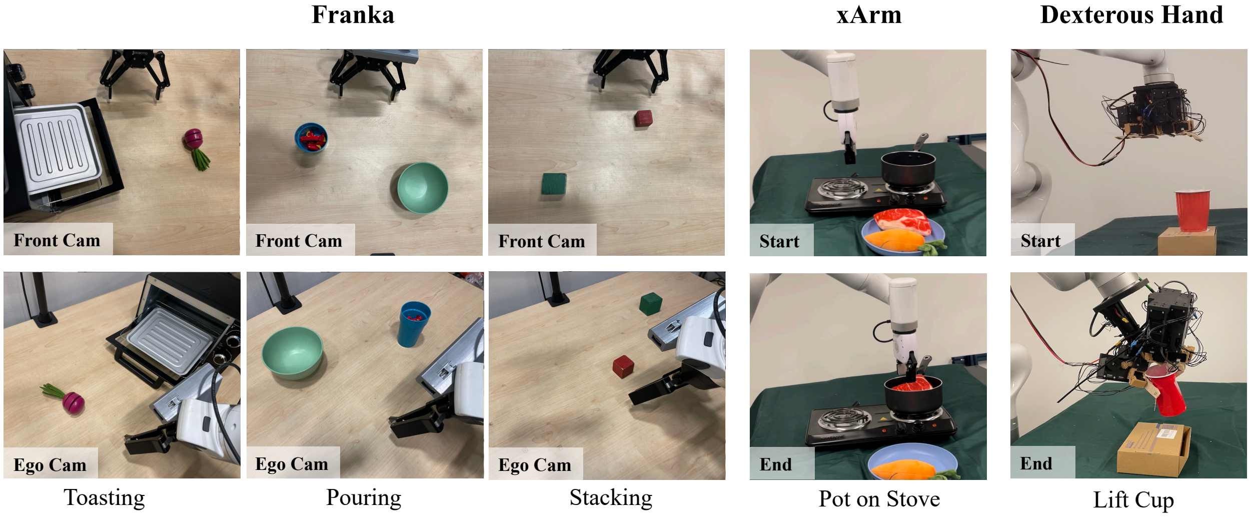

Figure 2.5: Our experiments consider 5 unique manipulation tasks, ranging from classic block-stacking to a multi-stage toasting scenario. These tasks are implemented on 3 unique robot setups, including a high Degree-of-Freedom dexterous hand (right). The 3 camera views shown – front, ego, and side views (for xArm/dexterous hand) – are the same views ingested by the policy during test-time. Note that 3 of the tasks consider 2 unique camera views in order to test for robustness!

actions, object bounding boxes, or motion forecasting. We observe that most tasks of interests humans perform are with their hands. We thus focus on training our model to predict hand-object interactions and hand motion.

We present HRP, a simple and effective representation learning approach that injects hand-object interaction priors into a self-supervised network, f_θ, using an automatically generated human affordance dataset, $\mathcal{D}_H$ (see above for definitions and dataset mining approach). HRP is illustrated in Fig. 2.2, and the following sections describe its implementation in detail.

2.4.1 Training HRP

The initial network f_θ is fine-tuned using batches sampled from the human dataset: $(I_t, c_t, h_t, b_t) \sim \mathcal{D}_H$, where c_t, h_t, and b_t are contact, hand, and object affordances corresponding to image I_t (see Sec. 2.3.2 for definitions). Some frames may not include all 3 affordances, so we include 3 mask variables – $m_t^{(c)}, m_t^{(h)}, m_t^{(b)}$ – so the missing values can be ignored during training. We add 3 small affordance modules – p_c, p_h, p_b – on top of f_θ that are trained to regress the respective affordances for I_t. This results in the following three loss functions:

$$\mathcal{L}_{\text{ct}} = ||c_t - p_c(f_\theta(I_t))||_2 \tag{2.1}$$

$$\mathcal{L}_{\text{hand}} = ||h_t - p_h(f_\theta(I_t))||_2 \tag{2.2}$$

$$\mathcal{L}_{\text{obj}} = ||b_t - p_b(f_\theta(I_t))||_2 \tag{2.3}$$

The full loss is:

$$\mathcal{L} = m_t^{(c)} \lambda_{\text{ct}} \mathcal{L}_{\text{ct}} + m_t^{(h)} \lambda_{\text{hand}} \mathcal{L}_{\text{hand}} + m_t^{(b)} \lambda_{\text{obj}} \mathcal{L}_{\text{obj}} \tag{2.4}$$

Where the λs are hyper-parameters that control the relative weight of each affordance loss. We empirically found $\lambda_{\text{obj}} = 0.05$, $\lambda_{\text{ct}} = 0.005$, $\lambda_{\text{hand}} = 0.5$ to be optimal for downstream performance (see Appendix **??**).

2.4.2 Implementation Details

Our affordance dataset ($\mathcal{D}_H$) is at least an order of magnitude smaller than the pre-training image dataset initially used by the baseline representation (e.g., ImageNet has 1M frames v.s. our 150K). To preserve the useful features learned from the larger pre-training distribution, we keep most of the parameters in θ fixed during HRP fine-tuning. Specifically, we only fine-tune the baseline network's normalization layers and leave the rest fixed, which has been shown to be an effective approach [99, 100]. In the case of our ViT-B this amounts to fine-tuning only the LayerNorm parameters γ and β:

$$\text{LayerNorm}(x) = \frac{x - \mu}{\sigma} \gamma + \beta \tag{2.5}$$

These parameters are fine-tuned to minimize $\mathcal{L}$ using standard back-propagation and the ADAM [101] optimizer.

2.5 Experimental Details

Our contributions are validated using a simple empirical formula: first, HRP is applied to each baseline model (listed below). Then, (following standard practice [17, 19, 14]) the resulting representation is fine-tuned into a manipulation policy using behavior cloning. Details for each stage are provided below, and the HRP is illustrated in Fig. 2.2.

Baseline Representations We chose 6 representative, SOTA baselines from both the vision and robotics communities:

1. **ImageNet MAE** was pre-trained by applying the Masked Auto-Encoders [24] (MAE) algorithm to the ImageNet-1M dataset [27]. It achieved SOTA performance across a suite of vision tasks, and is the first self-supervised representation to beat supervised pre-training. We use the standard Masked Auto Encoder training scheme for this, using hyperparmaeters from MAE [24].

2. **Ego4D MAE** was trained by applying the MAE algorithm to a set of 1M frames sampled from the Ego4D dataset [1]. For consistency with prior work, we use the same 1M frame-set sampled by the R3M authors [17]. We use the standard Masked Auto Encoder training scheme for this, using hyperparmaeters from MAE [24].

3. **CLIP [102]** is a SOTA representation for internet data. It was learned by applying contrastive learning [23] to a large set natural language - image pairs crawled from internet captions. We used publicly available model weights.

4. **DINO [29]** was trained using a self-distillation algorithm that encourages the network to learn local-to-global image correspondences. DINO's emergent segmentation capabilities could be well suited for robotics, and it has already shown SOTA performance in sim [15]. We used publicly available model weights.

5. **MVP [18]** was trained by applying MAEs to a mix of in-the-wild datasets (100 DoH [2], Ego4D [1], etc.). The authors showed strong performance on various manipulation tasks. We used publicly available model weights.

6. **VC-1 [19]** was trained in a similar fashion to MVP, but used a larger dataset mix. It showed strong performance on visual navigation tasks. We used publicly available model weights.

Note that each baseline is parameterized with the same ViT-B encoder w/ patch size 16 (see Sec. 2.3.2), to ensure apples-to-apples comparisons.

Policy Learning Each representation is evaluated on downstream robotic manipulation tasks, by fine-tuning it into a policy (π) using Behavior Cloning [36, 103, 104]. Note that π must predict the expert action (a_t – robot motor command) given the observation (o_t – input image and robot state): $a_t \sim \pi(\cdot|o_t)$. And π is learned using a set of *50 expert demonstrations* $\mathcal{D} = \{\tau_1, \dots, \tau_{50}\}$, where each demonstration $\tau_i = [(o_0, a_0), \dots, (o_T, a_T)]$ is a trajectory of expert observation-action tuples. In our case, π is parameterized by a small 2-layer MLP (p) placed atop the pre-trained encoder $p(f(o_t))$ that predicts a Gaussian Mixture policy distribution w/ 5 modes. Both the policy network and visual encoder are optimized end-to-end (using ADAM [101] w/ $lr = 0.0001$ for 50K steps) to maximize the log-likelihood of expert actions: $\max_{p,f} log(\pi(a_t|p(f(o_t))))$. During test time actions are sampled from this distribution and executed on the robot: $a_t \sim \pi(\cdot|p(f(o_t)))$. This is a standard evaluation formula that closely follows best practices from prior robotic representation learning work [4, 14].

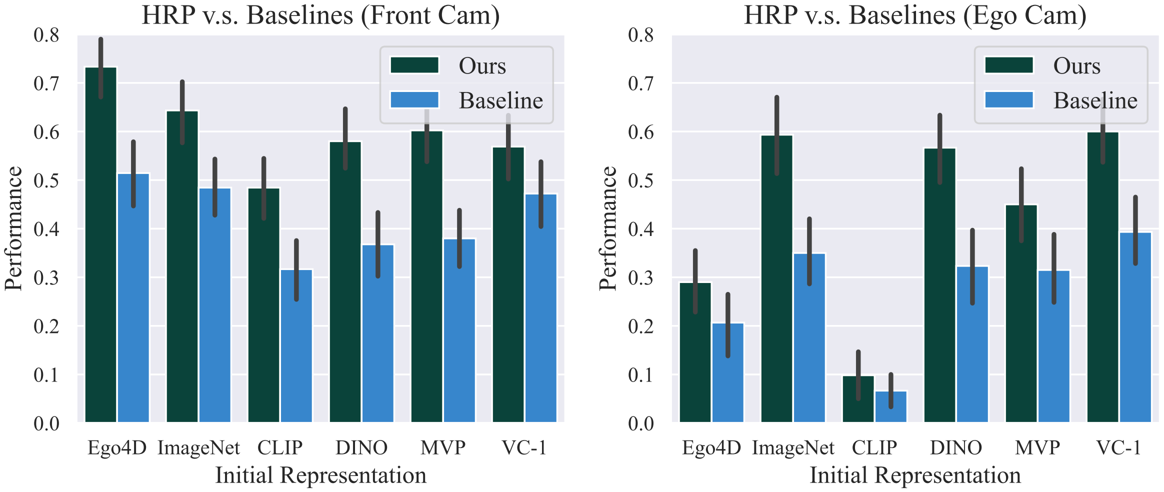

Figure 2.6: We apply HRP to 6 different baseline representations and plot how it affects performance on average across the *toasting*, *pouring*, and *stacking* tasks. This evaluation procedure is repeated using two distinct cameras (shown in Fig. 3.3) in order to test if HRP representation are robust to view shifts. We find that HRP representations consistently and substantially outperform their vanilla baselines, and that this effect holds across both the front (left) and ego (right) cameras. In fact, our strongest representation – ImageNet + HRP– delivers SOTA performance on both views!

Real World Tasks We fine-tune policies for each representation on the 5 diverse tasks listed below, which are implemented on 3 unique robotic setups, including a dexterous hand (illustrated in Fig. 3.3). 50 expert fine-tuning demonstrations were collected for each task via expert tele-operation. Note that the stacking, pouring, and toasting tasks were *evaluated twice using different camera views* to test robustness!

- **Stacking:** The stacking task requires the robot to pick up the red block and place it on the green block. During test time both blocks' starting positions are randomized to novel locations (not seen in training). A trial is marked as successful if the robot correctly picks and stacks the red block, and half successful if the red block is unstably placed on the green block. This task is implemented on a Franka robot and uses both an Ego and Front camera viewpoint.

- **Pouring:** The pouring task requires the robot to pick up the cup and pour the material (5 candies) into the target bowl. During test time we use novel cups and bowls and place each in new test locations. This task's success metric is the fraction of candies successfully poured (e.g., $2/5$ candies poured $\rightarrow$ 0.4 success). This task was also implemented on the Franka using Ego and Front cameras.

- **Toasting:** The toasting task requires the robot to pick up a target object, place it in the toaster oven, and shut the toaster. This is a challenging, multi-stage task. During test time the object type, and object/toaster positions are both varied. A test trial is marked as successful if the whole task is completed, and 0.5 successful if the robot only successfully places the object. This is the final task implemented on Franka w/ Ego and Front camera views.

- **Pot on Stove:** The stove task requires picking up a piece of meat or carrot from a plate and placing it within a pot on a stove. During test time, novel "food" objects are used and the location is randomized. A trial is marked as successful if the food is correctly placed in the pot. This task is implemented on a xArm and uses the side camera view.
- **Hand Lift Cup** This task requires a dexterous hand to reach, grasp, and lift up a deformable red solo up. The hand's high dimensional action space ($\mathcal{R}^{20}$) makes this task especially challenging. A trial is marked successful if the cup is stably grasped and picked. This task is implemented on a custom dexterous hand using a side camera view.

2.6 Results

Our experiments are designed to answer the following:

1. **Can HRP improve the performance of the pre-trained baseline networks (listed above)**? Does the effect hold across different camera views and/or new robots? (see Sec. 3.5.1)
2. Our affordance labels are generated using off-the-shelf vision modules – **does distilling their affordance outputs into a representation (via HRP) work better than simply using those networks as encoders?** (see Sec. 2.6.2)
3. How does HRP compare against alternate forms of supervision on the same human video dataset? (see Sec. 2.6.3)
4. How important are each of the three affordance losses for HRP's final performance? And is it really best to only fine-tune the LayerNorms and leave the other weights fixed? (see Sec. 2.6.4)
5. Can HRP handle scenarios with OOD distractor objects during test time? (see Sec. 2.6.5)
6. Can HRP representations work with different imitation learning pipelines, like diffusion policy [3]? (see Sec. 2.6.6)

Note that all experiments were conducted on real robot hardware, and the models were all tested back-to-back (i.e., using proper A/B evaluation) using 50+ trials per model to guarantee statistical significance. Note that all of our figures and tables report success rates (sometimes averaged across

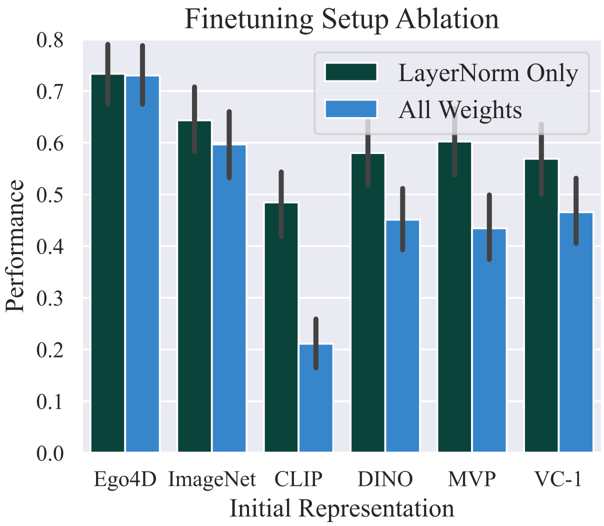

Figure 2.7: This chart applies an ablated HRP method (full fine-tuning) to the 6 baseline representations and compares their average performance v.s. standard HRP representations on the *toasting*, *pouring*, and *stacking* tasks (front cam). We find that LayerNorm only fine-tuning is almost always superior.

the toasting, stacking, and pouring tasks) alongside std. err. to quantify experimental uncertainty – i.e. success% ± std. err..

2.6.1 Improving Representations w/ HRP

To begin, we evaluate the 6 baseline representations (detailed in Sec. 2.5) on the *toasting*, *pouring*, and *stacking* tasks using the *front camera view*. Then, we apply HRP to each of these baselines, and evaluate those 6 new models on the same tasks. Average success rates across all 3 tasks are presented in Fig. 2.6 (left). First, this experiment demonstrates that ImageNet MAE is still highly competitive on real-world manipulation tasks when compared to other self-supervised representations from the vision [1, 29], machine learning [102], and robotics communities [13, 19]. Second, we show that HRP **uniformly boosts performance** on downstream robotics tasks – i.e., baseline + HRP > baseline for every baseline representation considered! Thus, we conclude that the affordance information injected by our method is highly useful for robot learning, and (for now) cannot be learned in a purely self-supervised manner.

Second Camera View A common critique is that robotic representations perform very differently when the camera view (even slightly) changes. To address this issue, we replicated the first experiment using a radically different *ego view*, where the camera is placed over the robot's shoulder (i.e., on its "head"). While perhaps a more realistic view, it is significantly more challenging due to the increased robot-object occlusion. Average success rates are presented in Fig. 2.6 (right).

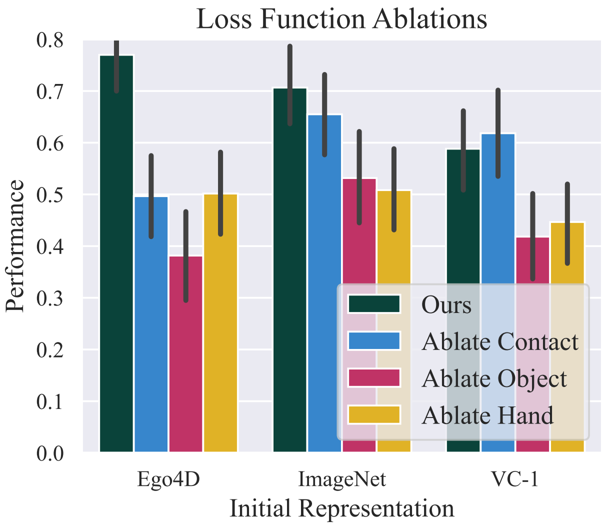

Figure 2.8: We drop each of the 3 losses in HRP, and compare the ablated method's average performance (across the *toasting*, *pouring*, *stacking* tasks) against full HRP representations. Due to the number of ablations involved, this experiment is only run on the Ego4D, ImageNet, and VC-1 base models. We find that the object and hand losses are critical for good performance, but the contact loss only makes a significant impact on the Ego4D base model.

Note that our findings replicate almost exactly from the front camera view. The ImageNet MAE representation is still competitive with the other baselines, and applying HRP uniformly improves the baseline performance. In addition, we find that **HRP injects a higher level of robustness to camera view shifts**, when compared to the baselines. For example, we find that ImageNet + HRP performs the same on the ego and front camera, even though the ImageNet baseline clearly prefers the front cam. This general effect holds (to varying degrees) across all six baselines!

Scaling to More Robots Finally, we verify that HRP representations can provide benefits on other robotic hardware setups. Specifically, we compare Ego4D + HRP and ImageNet + HRP versus the respective baselines on the *Pot on Stove* (xARM) and *Hand Lift Cup* (dexterous hand) tasks. Results are presented in Table 2.3. Note that HRP representations provide consistent and significant performance during policy learning on these radically different robot setups, which both also use a unique side camera view. This gives us further confidence in HRP's view robustness and demonstrates that these representations are not tied to specific hardware setups, and can scale to complex morphologies like dexterous hands.

2.6.2 Distillation w/ HRP Improves Over Label Networks

It is clear that applying HRP to self-supervised representations results in a consistent boost. However, the hand, object, and contact affordance labels for HRP themselves come from neural net-

Front Cam	**Teacher ResNet** 100DoH [2]	**HRP Models** w/ Ego4D	w/ ImageNet	w/ CLIP
Toasting	35% ± 15%	**83%** ± 9%	75% ± 10%	50% ± 11%
Pouring	34% ± 13%	**60%** ± 11%	48% ± 12%	39% ± 11%
Stacking	0%	**77%** ± 10%	70% ± 11%	57% ± 11%
Average	35% ± 10%	**73%** ± 6%	64% ± 7%	48% ± 6%

Table 2.1: This table compares 3 representations trained w/ HRP against the teacher ResNet [2] that generated our human affordance dataset (see Sec. 2.3.2). We find that the ResNet teacher under-performs even the worst HRP representation (fine-tuned from CLIP), *even after excluding the stacking task, which it failed on.*

	Ego4D		**ImageNet**		**CLIP**	
	+ HRP	+ Semantic	+ HRP	+ Semantic	+ HRP	+ Semantic
Toasting	**83%** ± 9%	25% ± 13%	**75%** ± 10%	40% ± 14%	**50%** ± 11%	20% ± 13%
Pouring	**60%** ± 11%	30% ± 13.4%	**48%** ± 12%	26% ± 11%	**39%** ± 11%	22% ± 10%
Stacking	**77%** ± 10%	30% ± 11%	**70%** ± 11%	40% ± 12%	**57%** ± 11%	30% ± 13%
Average	**73%** ± 6%	28% ± 7%	**64%** ± 7%	35% ± 7%	**48%** ± 6%	24% ± 7%

Table 2.2: We create Semantic representations by fine-tuning the Ego4D, ImageNet, and CLIP baselines using a classification loss, instead of HRP's affordance loss. Note that the exact same Ego4D clips (see Sec. 2.3.2) are used during semantic fine-tuning, thanks to object class labels generated automatically by Detic [105]. The sematic representations were evaluated (using the same BC pipeline) on the Toasting, Pouring, and Stacking tasks, and compared against their HRP counterparts. Success rates (and standard error) are reported above. We find that the affordance supervision provided by HRP is vastly superior to the semantic alternative.

works (see Sec. 2.3.2) – specifically we use the ResNet-101 [107] detector from 100DoH [2] as a label generator for our active object and contact affordance. The hand affordance we use comes from FrankMocap [28], which uses 100DoH [2] as a base model. Thus, does distilling labels from this detector via HRP actually provide a benefit over simply using the 100DoH model itself as a pre-trained representation? To test this question, we fine-tune policies on the toasting, pouring, and stacking (front cam) tasks and compare them against HRP applied to ImageNet, Ego4D, and (the weakest model) CLIP (see Table 2.1). In all cases, our representation handily beats the 100DoH policy. So while the affordance labels can dramatically boost policy learning (via HRP), the source/teacher models are not at all competitive on robotics tasks.

2.6.3 Comparing Against Alternate Forms of Supervision

We now analyze if HRP's losses are better suited for robotics tasks than an alternate supervision scheme. To be clear, the previous results already demonstrated that HRP + Ego4D out-performed the Ego4D baseline by up to 20% (see Fig. 2.6; left), despite being sourced from the same image

	Ego4D		**ImageNet**	
	w/ HRP	Baseline	w/ HRP	Baseline
Pot on Stove	**50%** ± 17%	40% ± 16%	**60%** ± 16%	40% ± 16%
Hand Lift Cup	**50%** ± 17%	40% ± 16%	**50%** ± 17%	30% ± 15%

Table 2.3: We present results of `Ego4D + HRP` and `ImageNet + HRP`, as well as the respective baselines on the x-Arm (Pot on Stove) and a dexterous hand task (Lift Cup). We see that `HRP` can even boost performance in multiple morphologies, including a high-degree of freedom dexterous hand [106].

Initialization	**w/ HRP**	**MAE Initialization**
Ego4D	**40%** ± 15%	15% ± 11%
ImageNet	**40%** ± 15%	**40%** ± 15%

Table 2.4: This table compares `Ego4D + HRP` and `ImageNet + HRP` representations against their respective baselines on a *stacking w/ distractors* task. Here the robot must successfully complete the usual stacking task, when extraneous objects (an orange carrot, and a green bowl) are added to the scene. We find that `Ego4D + HRP` improved over its baseline on this task, but `ImageNet + HRP` performed the same as its baseline.

data. However, it could be that the additional fine-tuning step with the $100K$ filtered interaction clips is responsible, and the specific affordance losses are not key. To test this, we ran a modified version of HRP using a semantic classification loss, instead of our affordance hand-object losses. The ground-truth labels for each image were obtained using the Detic object detector [105]. We then similarly fine-tuned the `ImageNet`, `Ego4D`, and `CLIP` baseline representation using these labels, and compared them against the respective `HRP` models on the toasting, pouring, and stacking tasks. The results are presented in Table 2.2 We find that the `HRP` models perform significantly better on every task. Thus, we conclude that HRP's affordance losses play an important role in boosting performance (i.e., it's not just data or extra fine-tuning).

2.6.4 What Design Decisions are Important?

The following section ablates the key components of `HRP` to evaluate their relative importance. First, we apply `HRP` to each of the 6 baseline representations again, but this time none of the weights are kept fixed (see Sec. 2.4.2). These representations are fine-tuned on the toasting, stacking, and pouring tasks (front cam), and compared against the original `HRP` representations in Fig. 2.7. Note that fine-tuning all the layers results in a substantial performance hit on average, and this trend is consistent regardless of the base representation! Thus, we conclude fine-tuning only the layer norms when applying `HRP` is the correct decision.

Next, we ablate each of the affordance losses in Eq. 2.4, by applying HRP three times: once with $\lambda_{\text{ct}} = 0$, then with $\lambda_{\text{hand}} = 0$, and finally $\lambda_{\text{obj}} = 0$. This process is repeated using 3 different base models; `ImageNet`, `Ego4D`, and `VC-1`. This creates 9 ablated models (3 losses x 3 ini-

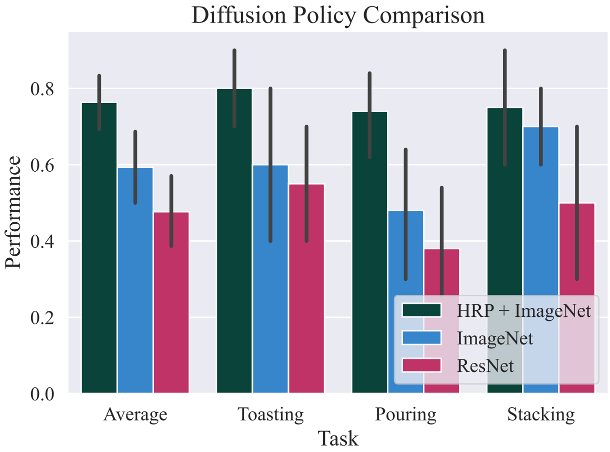

Figure 2.9: This figure tests if HRP representations can boost performance when using a radically different imitation learning framework – namely Diffusion Policy [3]. We evaluate diffusion policies (following the U-Net + state action formula described by Chi et. al [3]) on the toasting, pouring, and stacking tasks using 3 different visual encoders: the default ResNet encoder from RoboMimic [4], the `ImageNet + MAE` baseline, and our `HRP + ImageNet` features. We find a clear improvement when using HRP weights, which suggests that HRP is applicable to different imitation learning frameworks!

tializations) that are compared versus the full HRP models on the toasting, pouring, and stacking tasks. The average results are presented in Fig. 2.8. We find that removing the object (Eq. 2.3) and hand (Eq. 2.2) losses uniformly results in significant performance degradation. Meanwhile, the contact loss (Eq. 2.1) only provides a significant boost for the `Ego4D` base model but does not affect the others. Thus, we conclude that object and hand losses are critical for our method, while the contact loss is more marginal, most likely due to the fact that the extraction of contacts is a relatively noisy process.

2.6.5 Novel Distractors During Test-Time

We evaluate the performance of `HRP` and baseline approaches in OOD settings, by adding extraneous "distractor" objects (an orange carrot and a light green bowl) in the stacking task. The robot must successfully ignore the distractor and complete the task. Results are presented in Table 2.4. We found that both `ImageNet + HRP` and `ImageNet` had the same level of robustness to distractors. Meanwhile, `Ego4D`'s performance dropped substantially, while `Ego4D + HRP` remained robust. Our hypothesis is that human data by itself does not contain enough information to allow for OOD tasks. However, using `HRP` allows for more focus on task-relevant features, even when the representation is trained on less diverse data.

2.6.6 Evaluating w/ Diffusion Policy

Finally, we analyze if HRP representations offer improvements when using a radically different imitation learning framework, like diffusion policy [3]. Specifically, we adopt the original U-Net action prediction head and environment setup from Chi et. al. [3], but replace their ResNet visual encoder (inspired from RoboMimic [4]) with our HRP + ImageNet ViT-B model. Then we compare this HRP enhanced diffusion policy implementation, against (diffusion agents which use) both the original ResNet encoder and the baseline ImageNet ViT-B. Results for the (Franka) stacking, pouring, and toasting tasks are presented in Fig. 2.9. We find that HRP + ImageNet significantly improves over both alternatives (76% for HRP v.s., 56% for Chi et. al.'s implementation [3]), despite using a radically different imitation learning objective/setup! Thus, we conclude that HRP representations can boost performance across different setups.

2.7 Discussion

In HRP, we investigate human affordances as a strong prior for training visual representations. Thus, we present, a semi-supervised pipeline that extracts contact points, hand poses, and activate objects from human videos, and uses these affordances for fine-tuning representations. HRP improves base model performance drastically, for five different, downstream behavior cloning tasks, across three robot morphologies and three camera views. All components of our approach, including LayerNorm tuning, our three affordances, and our distillation process (from affordance labels to representations) are important for the model's success. One key limitation of this approach is that it has only been tested on imitation settings in this paper. In the future, we hope to not only scale this approach to many more tasks and robot morphologies, but also incorporate HRP in other robot learning paradigms such as reinforcement learning or model based control.

Part II

Dexterous Human Interactions for In-the-Wild Robot Policies

Chapter 3

DexWild: Dexterous Human Interactions for In-the-Wild Robot Policies

3.1 Introduction

Roboticists have long envisioned creating systems that exhibit the dexterity and adaptability of human behavior. Ideally, such robots would generalize across a wide range of objects, environments, and embodiments. However, this level of versatility remains a core challenge. Unlike recent advances in language [108, 109, 110] and vision-language models [111, 112], which are fueled by massive internet-scale datasets, robotics lacks data at comparable scale and diversity. This data scarcity makes it difficult to train general-purpose models or collect new data safely—creating a bottleneck that limits progress.

One common solution is teleoperation [43, 113, 114], where skilled operators generate high-quality demonstrations. While effective, these systems are expensive and time-consuming, often requiring specialized equipment and significant human effort. Data collection in diverse environments further compounds these issues, as it demands physically relocating hardware or replicating setups.

To address scalability, another line of work taps into internet videos [1, 115], which offer rich, diverse scenes of human interaction. Yet, these videos often lack the precision needed for fine-grained robotic learning, especially around hand-object interactions. While recent methods attempt to extract usable cues using pose or affordance models [116, 82, 117], these signals remain noisy and insufficiently grounded for training high-fidelity policies.

Wearable gripper systems [118, 119] offer more direct mapping to robotic control, but are cumbersome, fatiguing, and limited in their ability to capture natural human dexterity at scale.

In this paper, we introduce DexWild, a system that co-trains on both human and robot demon-

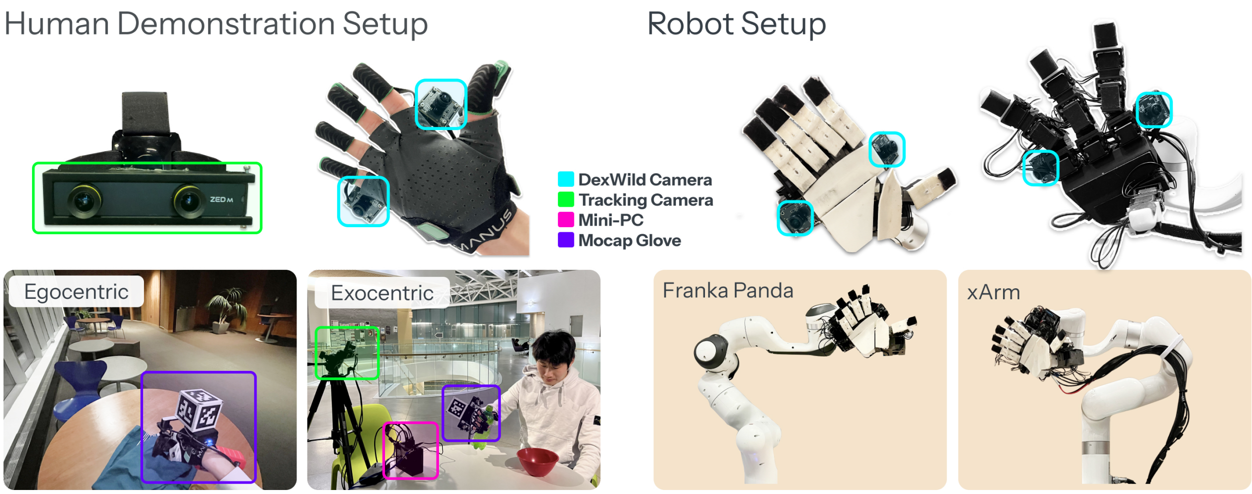

Figure 3.1: **Left:** DexWild efficiently capture high-fidelity data using an individual's own hands across various environments. **Right:** Robot hands are equipped with cameras aligned with the human cameras. We test DexWild on two distinct robot hands and robot arms.

strations to enable robust, generalizable manipulation policies. Our contributions include:

1. **Scalable Data Collection System**: We propose DexWild-System, a human-embodiment platform that enables 10 untrained users to collect 6,621 demonstrations across 66 environments, achieving a 4.6× speedup over robot-only collection.

2. **Efficient Co-training Framework**: We design a method for integrating human and robot data that improves policy generalization, yielding a 75.1

3. **Cross-Embodiment and Multi-Task Transfer**: Our system achieves an 8.3× improvement in cross-embodiment policy transfer and supports generalization across multiple tasks.

3.2 Related Works

3.2.1 Generalization for Imitation Learning

Learning generalizable policies for robot manipulation has seen rapid progress, driven largely by advances in visual representation learning and imitation learning from large-scale datasets. On the visual side, embodied representation learning has benefited from egocentric datasets such as Ego4D [1] and EPIC-KITCHENS [115], with recent methods [12, 14, 16, 120] leveraging these datasets to train scalable visual encoders. However, these approaches still require substantial downstream robot demonstrations to train control policies.

In parallel, robot-only demonstration datasets have grown significantly in scale and diversity [113, 43, 114], fueling research in behavior cloning and enabling generalist policy architectures [121, 43, 122]. While these policies show impressive performance across many tasks, they often struggle to generalize to unseen object categories, scene layouts, or environmental conditions [123]. This lack of robustness remains a key limitation of current systems.

3.2.2 Data Generation for Robot Manipulation

Overcoming the robot data bottleneck has become a central challenge in robot learning.

One approach leverages internet videos to extract action information. Several works, such as VideoDex [117] and HOP [124], utilize large scale human videos to learn an action prior through retargeting, which they use to bootstrap policy training. Others, such as LAPA [125], use unlabelled videos to generate latent action representations that can be used for downstream tasks. While these video-based schemes enjoy vast visual diversity, they typically fall short at capturing the precise, low-level motor commands needed for real-world manipulation.

Simulation enables rapid generation of action data at scale. However, creating diverse, realistic environments for many tasks and addressing the sim-to-real gap is challenging. Recent successes in transferring manipulation policies from simulation [126] have been confined to tabletop settings and lack the generalization needed for deployment in diverse environments.

Direct teleoperation on physical robots yields the highest fidelity, but scales poorly. Recent works have shown impressive dexterity and efficient learning in fixed scenarios [127, 128, 129, 130], yet collecting enough demonstrations to generalize across diverse scenes quickly becomes prohibitively expensive.

Recently, there has been a growing body of work that utilizes purpose-collected high quality human embodiment data without the tedious teleoperation. We discuss these approaches in the next section.

3.2.3 Human Action Tracking Systems

In order to acquire high-quality data from human motions, accurate hand and wrist tracking is of paramount importance. To bypass the complexities of hand pose estimation, several works equip users with handheld robot grippers [118, 131, 33]. While this approach simplifies retargeting, it constrains users to the specific morphology of the robot gripper, limiting the diversity of captured behavior. Moreover, many of these systems rely on SLAM-based wrist tracking, which can fail in feature-sparse environments or when occlusions occur [118, 132]—such as during drawer opening or tool use.

Other approaches aim to estimate both hand and wrist poses directly from visual input [133, 134, 135, 136, 137, 138, 139]. These methods are easy to deploy and require no instrumentation, but their performance degrades significantly under occlusion—an unavoidable situation in manipulation. Alternative strategies for wrist tracking, such as IMU-based [140, 141] and outside-in optical systems [142], come with their own limitations: IMUs are lightweight and portable but prone to drift, while optical systems are accurate yet require laborious calibration and controlled environments.DexWild leverages calibration-free Aruco tracking—significantly improving reliability and minimizing setup time as it requires a single monocular camera.

While vision-based methods often attempt to track both the wrist and fingers simultaneously, many recent systems decouple the two to improve accuracy. Kinematic exoskeleton gloves can provide high-fidelity joint measurements and even haptic feedback [143], but are bulky and uncomfortable for long-term use. Instead, DexWild, along with prior works [129, 119], adopts a lightweight glove-based solution that uses electromagnetic field (EMF) sensing to estimate fingertip positions. This allows for accurate, real-time hand tracking that is robust to occlusions and readily retargetable to a wide range of robot hands.

3.3 DexWild

Many believe that leveraging large, high-quality datasets is the key for creating dexterous robot policies that generalize [43, 121, 117, 14]. We introduce DexWild-System, a user-friendly, high-fidelity platform for efficiently gathering natural human hand demonstrations across diverse real-world settings. Compared to traditional teleoperation-based approaches, DexWild-System enables 4.6× faster data acquisition at scale.

Building on this system, we propose DexWild, an imitation learning framework that co-trains on large-scale DexWild-System human demonstrations alongside a small number of robot demonstrations. This approach combines the diversity and richness of human interactions with the grounding of the robot embodiment, enabling policies to robustly generalize across new objects, environments, and embodiments. Figure 2.1 displays our high level approach.

3.3.1 Data Collection System

A scalable data collection system for dexterous robot learning must enable natural, efficient, and high-fidelity collection across diverse environments. To this end, we design DexWild-System: a portable, user-friendly system that captures human dexterous behavior with minimal setup and training. While previous in-the-wild data collection approaches have typically relied on sensorized grippers, we aimed to create a more intuitive hardware interface that mirrors how humans naturally interact with the world. From delicate fine-motor actions to powerful grasps, humans possess dexterity across a wide range of manipulation tasks. By learning from this intrinsic capability, DexWild-System captures rich, diverse data applicable to a broad range of robot embodiments.

DexWild-System is designed around three core objectives:

- **Portability:** Allow rapid, large-scale data collection across diverse environments without requiring complex calibration procedures.
- **High Fidelity:** Accurately capture fine-grained hand and environment interactions essential for training precise dexterous policies.
- **Embodiment-Agnostic:** Enable seamless retargeting from human demonstrations to a wide variety of robot hands.

Portability:

To collect data in diverse real-world settings, a system must be portable, robust, and usable by anyone. We design DexWild-System with these goals in mind: it is lightweight, easy to carry, and can be set up in just a few minutes—enabling scalable data collection across many locations.

As shown in Figure 3.1, DexWild-System consists of only three components: a single tracking camera for wrist pose estimation, a battery-powered mini-PC for onboard data capture, and a custom sensor pod comprising a motion-capture glove and synchronized palm-mounted cameras.

Unlike traditional motion capture systems [144, 145, 146, 147] that often rely on complex outside-in tracking setups that require calibration, DexWild-System is truly calibration free, making it versatile for any scenario and foolproof for untrained operators.

This is achieved by adopting a relative state-action representation, where each state and action is captured as the relative difference from the previous time step's pose. This eliminates any need for a global coordinate frame, allowing the tracking camera to be freely placed—either egocentrically or exocentrically. Additionally, the palm cameras are rigidly mounted in fixed positions across both human and robot embodiments. This ensures visual observations are aligned across domains, eliminating the need for further calibration at deployment. The external tracking camera, when carefully positioned, can also capture supplementary environmental context useful for learning robust policies.

High Fidelity:

To learn dexterous behaviors, fine-grained, nuanced motions must be captured in the training dataset. Although DexWild-System consists of only a few portable components, we make no compromises on data fidelity. Our system is designed to accurately capture both hand and wrist actions, paired with high-quality visual observations.

For wrist and hand tracking, vision-only methods are easy to setup. However, what they gain in portability, they often lose in accuracy and robustness—yielding noisy pose estimates that degrade policy learning [129, 148, 139, 118].

For hand pose estimation, we use motion capture gloves, which offer high accuracy, low latency, and robustness against occlusions [129]. For wrist tracking, we mount ArUco markers on the glove and track them using an external camera. This avoids the fragility of SLAM-based wrist tracking, which often fails in feature-sparse environments or during occlusion-heavy tasks (e.g., drawer opening).

Unlike many datasets that rely on egocentric or distant external cameras, we place two global-shutter cameras directly on the palm. As illustrated in Figure 3.1, these stereo cameras capture detailed, localized interaction views with minimal motion blur and a wide field of view. This wide field of view enables policies to operate using only the onboard palm cameras, without any reliance on static viewpoints.

Embodiment-Agnostic:

To ensure the longevity and versatility of DexWild data, we aim for it to remain useful across different robot embodiments—even as hardware platforms evolve. Achieving this goal requires careful alignment of both the observation space and the action space between humans and robots.

We begin by standardizing the observation space. Although our palm-mounted cameras have a wide field of view, we intentionally position them to focus primarily on the environment, minimizing the visibility of the hand itself. Importantly, the camera placement is mirrored between the human and robot hands. As shown in Figure 3.2, this design yields visually consistent observations across embodiments, allowing the policy to learn a shared visual representation that generalizes across both human and robot domains.

For action space alignment, we build on insights from prior work [149, 150], optimizing robot hand kinematics to match the fingertip positions observed in human demonstrations. We note that this method is general and can work for any robot hand embodiment. It operates with fixed hyperparameters across users and is robust to variations in hand size—eliminating the need for user-specific tuning.

Collecting data using natural human hands offers benefits beyond ease of use. The diversity in hand morphology across human demonstrators introduces useful variation, which we hypothesize helps policies learn more generalizable grasping strategies—particularly important given the inherent mismatch between human and robot hand kinematics.

In summary, DexWild is a portable, high quality, human-centric system that can be worn by any operator to collect human data in real-world environments. Next, we explain how we use the data collected by DexWild to enable dexterous policies to generalize to in-the-wild scenarios.

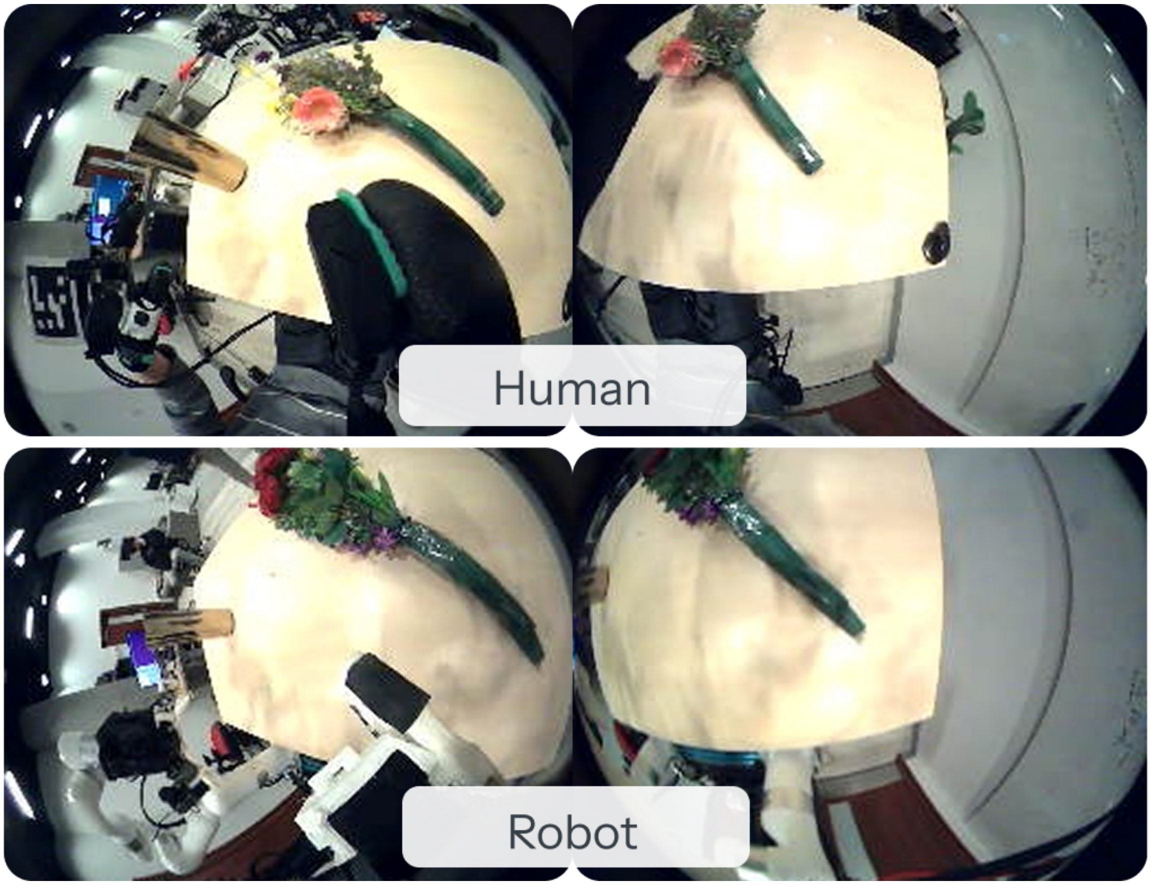

Figure 3.2: DexWild aligns the visual observations between humans and robots to bridge the embodiment gap. This incentivizes the model to learn a task-centric rather than embodiment-centric representation.

3.3.2 Training Data Modalities and Preprocessing

Generalization in dexterous manipulation demands both scale and embodiment grounding. With this goal, DexWild collects two complementary datasets: a large-scale human demonstration dataset D_H using DexWild-System, and a smaller teleoperated robot dataset D_R. Human data offers broad task diversity and ease of collection in real-world settings, but lacks embodiment alignment. Robot data, while limited in scale, provides crucial grounding in the robot's action and observation spaces. To harness the strengths of both, we co-train policies using a fixed ratio of human and robot data within a batch, (w_h, w_r)—balancing diversity with embodiment grounding to enable robust generalization during deployment.

At each training iteration, we sample a batch consisting of transitions x_h and x_r from D_H and D_R, respectively, according to the co-training weights. Each transition x_i at timestep i contains:

- **Observation** o_i: An observation at a given timestep consists of two synchronized palm camera images I_{pinky} and I_{thumb} captured at the current timestep, as well as a sequence of histori-

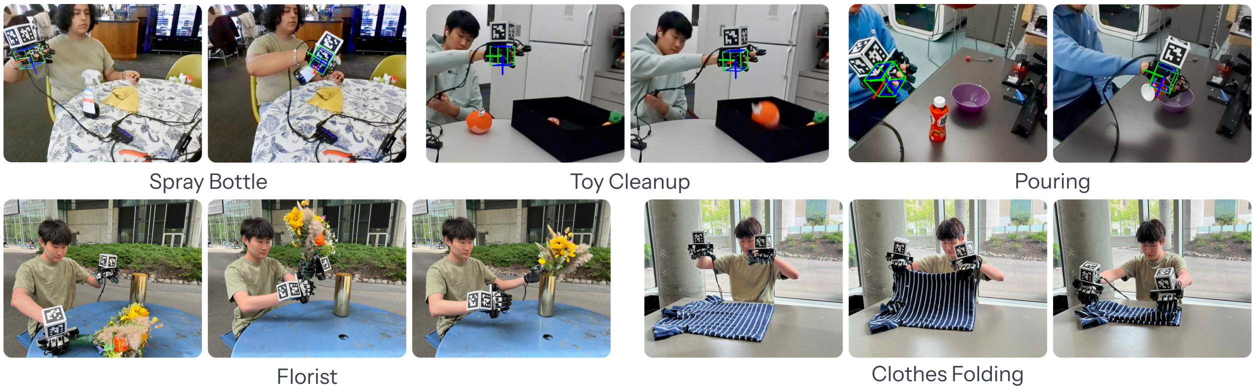

Figure 3.3: Using DexWild-System, humans can effortlessly collect accurate data with their own hands across a wide range of environments. This data is directly used to train any robot hand to perform dexterous manipulation in a human-like way in any environment. We validate this approach on five representative tasks. Please see videos of these tasks on our website at https://dexwild.github.io

cal states, sampled at a step size up a given horizon H, comprising of $\{\Delta p_i, \Delta p_{i-\text{step}}, ..., \Delta p_{i-H}\}$. Each Δp consists of relative historical end-effector positions.

- **Action** $a_{i:i+n-1}$: An action chunk of size n that includes actions $\{a_i, a_{i+1}, \ldots, a_{i+n-1}\}$, where a_i is the action at the current timestep. Specifically, a_i is a 26-dimensional vector consisting of:
  - a_{arm}: A 9-dimensional vector describing relative end-effector position (3D) and orientation (6D).
  - a_{hand}: A 17-dimensional vector describing the finger joint position targets of the robot hand.

  For bimanual tasks, the observation and action spaces are duplicated, and the inter-hand pose is appended to the observation to facilitate coordination.

While our retargeting procedure brings human and robot trajectories into a shared action space, a few additional steps are necessary to make the human and robot datasets compatible for joint training:

- **Action Normalization**: The actions of human and robot data are normalized separately to account for inherent distribution mismatches.
- **Demo Filtering**: Since human demonstrations are collected by untrained operators in uncontrolled environments, we apply a heuristic-based filtering pipeline to automatically detect and remove low-quality or invalid trajectories. This filtering step significantly improves dataset quality without manual labeling.

3.3.3 Policy Training

Through the careful design of our hardware, observation, and action interfaces, we are able to train dexterous robot policies using a simple behavior cloning (BC) objective [36, 103, 104]. To effectively learn from our multimodal, diverse data, our training pipeline leverages large-scale pre-trained visual encoders and shows strong performance across different policy architectures.

Visual Encoder: Training on DexWild data exposes our policy to significant visual diversity—across scenes, objects, and lighting—requiring an encoder that generalizes well to such variability. To address this, we adopt a pre-trained Vision Transformer (ViT) backbone, which has shown superior performance over ResNet-based encoders on in-the-wild manipulation tasks [151, 132]. Pre-trained ViTs, especially those trained on large internet-scale datasets, are particularly effective at extracting rich, transferable features [12, 152, 16, 14], making them well-suited for our setting.

Policy Class: While several imitation learning architectures have been proposed recently [127, 3], we adopt a diffusion-based policy. Diffusion models are particularly well-suited for dexterous manipulation, as they can capture multi-modal action distributions more effectively than alternatives such as Gaussian Mixture Models (GMMs) or transformers. This capability becomes increasingly important in DexWild, where demonstrations are collected from multiple humans with diverse strategies, resulting in inherently multi-modal behaviors. As the dataset scales, modeling this variability becomes critical for robust policy learning. Specifically, DexWild uses a diffusion U-Net model [3] to generate action chunks.

Concretely, the training procedure is outlined in Algorithm 1.

Algorithm 1 DexWild Imitation Learning Procedure

```
Require: Human dataset $\mathcal{D}_H$, Robot dataset $\mathcal{D}_R$, Co-training weights $\{\omega_h, \omega_r\}$
Initialize policy $\pi_\theta$ with ViT encoder $\phi_{\text{vit}}$
while not converged do
    Sample a batch of transitions $\{x_h\}, \{x_r\}$ from $\mathcal{D}_H, \mathcal{D}_R$ using weights $\{\omega_h, \omega_r\}$
    for each transition $x_i$ in the batch do
        Extract observation $o_i$
        Encode images: $Z_i = \phi_{\text{vit}}(o_i)$
        Extract ground truth action chunk $a_{i:i+n-1} = \{a_i, \dots, a_{i+n-1}\}$
        Sample noise scale $t \sim \mathcal{U}(1, T)$
        Add noise $\epsilon_t \sim \mathcal{N}(0, \sigma_t)$ to $a_{i:i+n-1}$
        Predict noise $\hat{\epsilon}_\theta = \pi_\theta(Z_i, a_{i:i+n-1} + \epsilon_t, t)$
        Compute diffusion loss $\mathcal{L}_\theta = \|\epsilon_t - \hat{\epsilon}_\theta\|_2^2$
    end for
    Update policy parameters $\theta$
end while
```

An important finding in our training framework is that tuning the human-to-robot data weight-

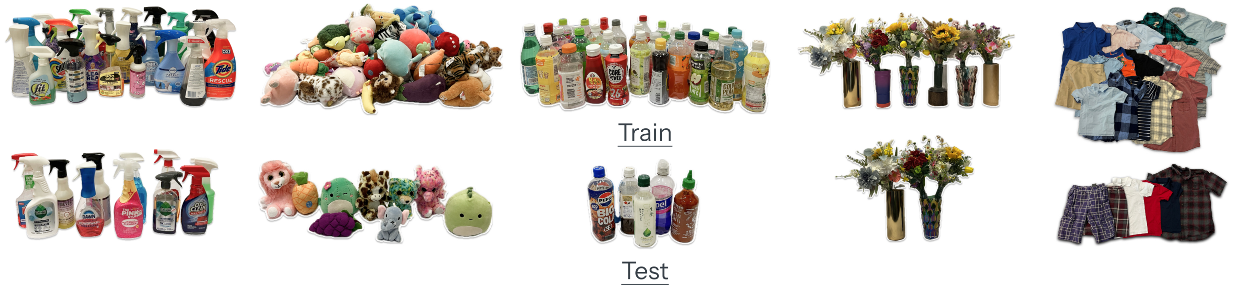

Figure 3.4: We collect data using a diverse set of objects across categories. *Spray Bottle Task* – 25 Train, 11 Test; *Toy Cleanup Task* – 64 Train, 9 Test; *Pour Task* – 35 Train, 5 Test; *Florist Task* - 6 Train, 2 Test; *Clothes Folding Task* - 17 Train, 6 Test.

ing significantly affects real-world performance. We discuss these effects in Section 3.5.1.

3.4 Experiments

Our experimental evaluation encompasses extensive real-world deployment across diverse environments and robots, utilizing both human demonstrations and robot teleoperation data. Below, we outline our data collection process, experimental setup, and evaluation tasks.

3.4.1 Scaling up Data Collection

Our hardware system was deployed to 10 untrained users to collect data across a wide range of real-world environments. These settings included indoor and outdoor locations, day and night conditions, crowded cafeterias and quiet study areas, with varied tables, objects, and lighting setups. The collectors themselves varied in hand sizes and demonstration styles, enabling us to learn from a wide distribution of environments and interactions.

We constructed two datasets through our collection efforts: D_H (human-collected data) and D_R (robot-collected data). The human dataset D_H comprises 9,290 demonstrations across five tasks: 3,000 demonstrations from 30 different environments for each of the *Spray Bottle* and *Toy Cleanup* tasks, 621 trajectories from 6 environments for the *Pour* task, 1,545 demonstrations from 15 environments for the *Florist* task, and 1,124 demonstrations from 12 environments for the *Clothes Folding* task.

The robot dataset D_R includes 1,395 demonstrations: 388 for *Spray Bottle*, 370 for *Toy Cleanup*, 111 for *Pour*, 236 for *Florist*, and 290 for *Clothes Folding* tasks. Robot data was collected using an xArm and LEAP hand V2 Advanced. Our training and test objects are detailed in Figure 3.4.

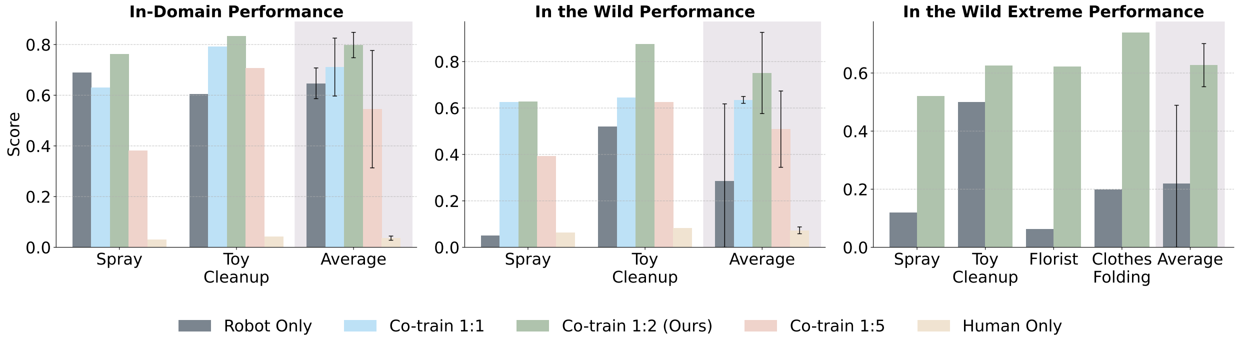

Figure 3.5: **How does co-training help with scaling up in the wild performance?** We evaluate our policy across three scenarios: (a) In-Domain scenes where robot training data was collected but with novel objects, (b) In-the-Wild scenes present in DexWild but not in robot data, and (c) In-the-Wild Extreme scenes absent from both datasets. Displayed ratio is Robot:Human.

3.4.2 Evaluation Tasks

We evaluate our approach on five diverse manipulation tasks, each designed to assess specific aspects of dexterous manipulation: functional grasping, long-horizon planning, cross-task transfer, bimanual coordination, and deformable object manipulation. A task visualization is provided in Figure 3.3.

In the Spray Bottle task, the robot grasps a spray bottle by the handle and sprays a target cloth, testing functional grasping and affordance understanding. In Toy Cleanup, the robot picks up scattered toys and places them in a bin, evaluating generalization and long-horizon planning. The Pouring task involves tilting a bottle to pour into a container, demonstrating skill transfer from the spray bottle task. In Bimanual Florist, the robot hands over a flower between its arms and inserts it into a vase, testing precise bimanual coordination. Finally, in Bimanual Clothes Folding, the robot uses both hands to fold a clothing item, assessing manipulation of deformable objects.

These tasks systematically evaluate HRP *functional grasping* capabilities, *generalization* across object types, *transferal* of skills across tasks, *coordination* between arms, and *adaptability* to deformable objects. Success requires the policy to adapt to varying object properties, environmental conditions, and task constraints.

3.4.3 Evaluation Environments

For robot experiments, we employed an xArm robot and Franka system, both equipped with either LEAP hand or LEAP hand V2 Advanced [**?** 129]. Unless explicitly mentioned, xArm and LEAP hand V2 Advanced was used. We evaluate our approach across three scenarios:

1. In-Domain: Environments where robot training data was collected, testing with novel objects
2. In-the-Wild: Environments present in DexWild but absent from robot training data

3. In-the-Wild Extreme: Unseen environments absent from both datasets.

3.5 Analysis and Results

In our evaluations, we seek to investigate the following key questions:

1. How effectively does DexWild leverage human data to achieve strong in-the-wild performance?

2. Does DexWild enable policy transfer across tasks and robot embodiments?

3. Does policy performance scale effectively with increasing amounts of DexWild-System data?

3.5.1 Zero Shot In the Wild Policies w/ DexWild

DexWild enables strong policy generalization in novel scenes. We evaluate policies in environments with increasing novelty to assess their generalization. As shown in Figure 3.5, policies trained exclusively on robot data perform well in in-domain settings (64.7% success rate) but degrade significantly in more challenging scenarios—in-the-wild (28.5%) and in-the-wild extreme (22.0%). This 36-point performance drop suggests that robot-only policies overfit to environment-specific features and fail to develop robust, transferable representations. In contrast, policies trained only on human data learn high-level object affordances and approach objects reliably, even in complex scenes. However, without robot-specific action grounding, they struggle to execute precise manipulation, resulting in poor performance across all scenarios (3.6% in-domain, 7.3% in-the-wild).

To combine the strengths of both modalities, we adopt a co-training strategy—jointly training on both robot and human data—a method validated in prior works [43, 121, 113, 138, 139]. This encourages the policy to learn task-relevant features rather than overfitting to specific embodiments or environments. We experiment with different **robot-to-human** data ratios (1:1 to 1:5) per training batch. Our empirical analysis reveals that a 1:2 ratio yields optimal performance across all scenarios:

1. In Domain: 79.8% vs. 64.7% (robot-only)

2. In-the-wild: 75.1% vs. 28.5% (robot-only)

3. In-the-wild Extreme: 62.7% vs. 22.0% (robot-only)

Interestingly, increasing the human data ratio further (e.g., 1:5) degrades performance (54.5% in-domain, 50.9% in-the-wild), indicating that robot data remains essential for grounding fine-grained control.

DexWild extends to complex bimanual coordination tasks. To evaluate whether DexWild generalizes beyond single-arm tasks, we test it on bimanual tasks that demand precise coordination between two hands. We compare co-trained policies (1:2 ratio) against robot-only policies in in-the-wild extreme settings. DexWild policies achieve a strong 68.1% average success rate, compared to just 13% for the robot-only baseline. Even when failures occur, DexWild policies exhibit meaningful attempts at task execution—while robot-only policies often produce erratic or unstructured behavior.

These results demonstrate that DexWild not only enables robust generalization across environments but also scales to more complex manipulation behaviors.

3.5.2 Robust Cross-Task and Cross-Embodiment Generalization

DexWild enables transfer of low-level skills across tasks. Many manipulation tasks share foundational motor skills—such as lifting, orienting, and rotating objects—which opens the door to skill reuse across related tasks. For example, opening a microwave and opening a cupboard both involve similar coordination and control. We evaluate this form of cross-task transfer using the *pouring* task, which shares many motion primitives with the *spray* task. Crucially, we use no robot data for pouring and instead combine human (DexWild-System) demonstrations of pouring with robot demonstrations from spraying. This setup enables **zero-shot generalization** to pouring in in-the-wild extreme environments. Using a 1:2 robot-to-human co-training ratio, our policy achieves a **94% success rate**, far exceeding policies trained with only robot (0%) or only human data (11%).

DexWild enables transfer across robot embodiments. Since DexWild data is not tied to any specific embodiment, it naturally supports cross-platform transfer. This prolongs the value of our data, as collecting platform-specific data for every new robot is resource-intensive and impractical. We test two transfer scenarios in in-the-wild extreme scenes:

- **Cross-arm**: Transferring from an xArm to a Franka Panda arm. We achieve a 37.5% success rate, compared to 4.5% for the robot-only baseline—an **8.3× improvement**.

- **Cross-hand**: Transferring from the LEAP Hand V2 Advanced to the original LEAP Hand. We achieve 65.3% success versus 13.3% for the baseline, showing that DexWild generalizes not only across arms, but across dexterous hands as well.

These results, shown in Figure 3.6, demonstrate that DexWild enables zero-shot generalization to new tasks and hardware embodiments **without any additional robot-specific data**, making it

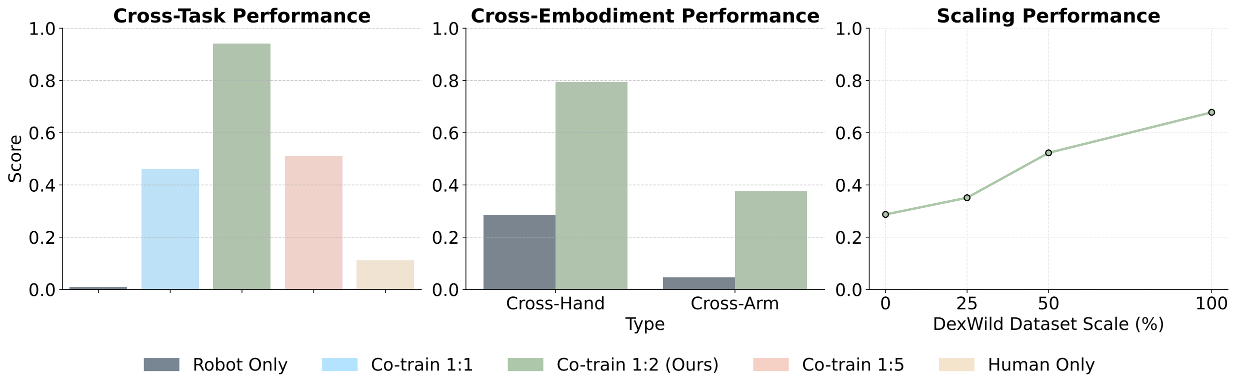

Figure 3.6: Left: **Cross-Task Performance** – Evaluating DexWild on the pour task using robot data exclusively from the spray task. Middle: **Cross-Embodiment Performance** – Testing DexWild policy on the Original LEAP hand and a Franka robot arm. Right: **Scaling Performance** – Demonstrating improved DexWild performance as dataset size increases. Displayed ratio is Robot:Human.

an efficient and general framework for dexterous policy learning on many robots.

3.5.3 Scalability of DexWild

Policy performance scales with dataset size. To understand how data scale impacts policy performance in the wild, we randomly sample subsets of the full human dataset at varying sizes and evaluate the resulting policies. We fix the size of the robot dataset. As shown in Figure 3.6, there is a clear positive correlation between dataset size and average task performance—rising from 28.7% at 20% dataset size to 67.8% with the full dataset, marking a 2.36× improvement. Interestingly, the learning curve is nonlinear, with especially steep gains in the 25–50% range, suggesting a critical threshold where the policy begins to reliably learn generalizable behaviors.

Importantly, performance continues to improve all the way to 100% data usage, indicating that

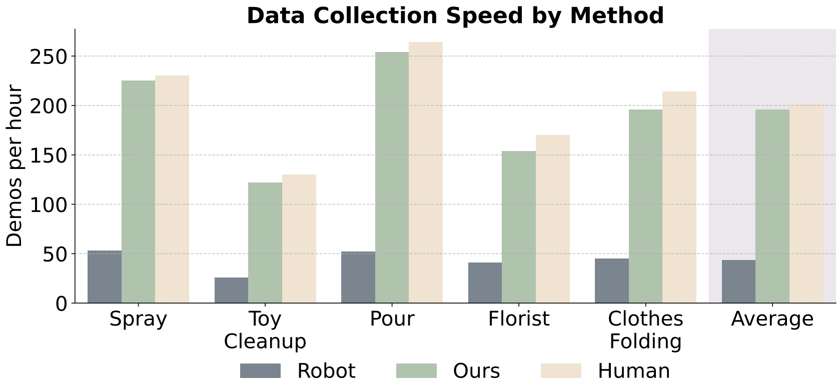

Figure 3.7: DexWild-System offers **4.6**× improvement over robot data collection speed and nearly matches the human bare hands data collection speed.

the system has not yet plateaued. This suggests that even more capable policies could be learned with continued data collection.

DexWild-System enables fast and scalable data collection. Given the observed benefits of scaling, we evaluate the data collection efficiency of DexWild-System via a comparative user study measuring demonstrations per hour. As shown in Figure 3.7, DexWild-System achieves an average collection rate of **201 demos/hour** across five representative tasks—nearly matching the rate of demonstrations collected using bare hands and **4.6× faster** than a traditional robot teleoperation system based on Gello [129, 128], which achieves just 43 demos/hour.

We identify three key limitations of Gello-based collection that our system overcomes:

1. **Lack of haptic feedback:** Operators cannot feel objects, making fine manipulation difficult for certain tasks.

2. **Scene reset:** Resetting the environment is cumbersome and often requires a second operator or pauses in data collection.

3. **Hardware setup overhead:** Robots are heavy and require time-consuming setup at each new location, whereas DexWild-System is portable and can be set up in minutes.

DexWild not only demonstrates strong scaling trends with increasing data volume, but also offers a practical and efficient path to collecting diverse, high-quality data at scale—crucial for real-world generalization.

3.6 Conclusion and Limitations

We introduce DexWild, a scalable framework for learning dexterous manipulation policies that effectively generalize to new tasks, environments, and robot embodiments. We introduce DexWild-System, a portable, human-centric data collection device that significantly accelerates dataset creation (4.6× faster than conventional robot teleoperation). We propose DexWild cotraining method, which leverages large scale human demonstrations alongside minimal robot data to achieve robust generalization-reaching a success rate of 68.5% in completely unseen environments, nearly four times higher than methods using robot data only. Furthermore, DexWild's embodiment-agnostic design enables strong cross-embodiment and cross-task transfer capabilities, reducing the need for robot-specific data.

Despite these strengths, several limitations remain that motivate future research: First, our approach still depends on a limited number of teleoperated robot data to bridge the gap between human and robot actions. Future work could explore improved retargeting or online policy adaptation to remove the need for teleoperated data. Next, because humans typically perform these tasks successfully, their demonstrations seldom include error recovery—causing trained policies to struggle to recover from unexpected failures. Adding recovery examples or adaptive strategies could boost real-world robustness. Finally, our method uses only visual and kinematic data, which limits its performance in contact-rich tasks. Incorporating tactile or haptic sensing could improve the handling of delicate interactions.

In summary, DexWild represents a significant step toward scalable, generalizable robot manipulation policies. Our results highlight the promise of leveraging human interaction data at scale, offering an exciting avenue toward truly dexterous and versatile robots operating in diverse, real-world environments.

Chapter 4

Conclusion and Future Directions

In this thesis, we explored how human manipulation priors—both passive and active—can be systematically leveraged to enable generalization in robotic learning. Part I introduced HRP, a framework that distills structured knowledge from passive human video into visual representations suitable for robot control. Part II introduced DexWild, a system for collecting and co-training on large-scale in-the-wild human demonstrations to drive policy generalization across tasks, environments, and embodiments. Together, these approaches offer complementary pathways toward the broader goal of *learning to generalize via human manipulation priors*. In this chapter, we summarize key contributions and highlight future directions that build on this foundation.

4.1 Summary of Contributions

Human Video Pre-Training (HRP): We showed that by mining affordance labels (hand-object contact points, hand poses, etc.) from large-scale human videos, we can pre-train a robot's visual encoder to be far more attuned to manipulation-relevant features. HRP-pretrained representations significantly improved downstream policy learning, boosting success across tasks and even in unseen conditions. This contribution is important because it demonstrates a way to leverage abundantly available *passive* human data (like YouTube videos or egocentric recordings) without requiring any manual labeling or robot data. HRP essentially gives robots a "head start" by learning what to pay attention to in a scene (e.g., where hands tend to grasp objects) before the robot even begins acting.

Wearable Human Demonstrations and Co-training (DexWild): We developed a practical system for capturing human demonstrations in the wild and a co-training algorithm that integrates this data with robot experiences. The resulting DexWild policies achieved unprecedented generalization: adapting to new environments, objects, and even different robot hardware with minimal to no extra training. This contribution addresses the active side of learning from humans—learning not just from observation but from *doing* as humans do. By utilizing human demonstrations, we circumvent the need for laborious teleoperation and tap into the intuition and efficiency of human motor skills to directly program robot behavior. The co-training approach ensures that this wealth of human skill is effectively translated into the robot's context.

Complementarity of Approaches: The two approaches tackle the generalization problem from different angles. HRP focuses on *representation learning* (perception), making sure the robot interprets its sensory inputs in a human-like way. DexWild focuses on *policy learning* (action), guiding the robot's decisions using human strategies. Together, they form a powerful combination: a robot could, for instance, use an HRP-pretrained visual encoder within a DexWild co-trained policy. We expect that HRP's visual understanding would enhance the policy's ability to perceive new scenarios, while DexWild's human-derived action strategies would enhance decision-making. An integrated system would effectively leverage human insight at both perception and control levels.

4.2 Future Directions

Building on this work, there are several exciting directions to pursue:

- **Unified Framework:** As noted, combining HRP and DexWild more explicitly is a logical next step. One could imagine a training pipeline where human videos not only pre-train the visual encoder but are also used (via imitation or inverse reinforcement learning) to suggest high-level goals or sub-tasks for the policy.

- **Scaling and Diversity:** Both methods likely benefit from more data. On the HRP side, incorporating more complex affordances (like tool affordances, or human intent prediction) could enrich the representation. On the DexWild side, expanding the dataset to more people (diverse sizes, techniques) and tasks will make policies even more robust. A potential future dataset could involve a crowd-sourced approach, where many users around the world contribute demonstrations with a standardized kit.

- **Interactive Fine-Tuning:** One limitation of imitation learning (used in both parts) is that it doesn't handle novel failures by the robot after training. A future direction is to allow the robot to fine-tune its skills through trial-and-error, *while still guided by human priors*. For example, after HRP+DexWild training, the robot could perform reinforcement learning on a new task; since it already has good representations and priors, it should learn much faster (this could be tested and formalized as accelerating RL with human priors).

- **Multi-Modal Integration:** Our HRP approach focused on visual inputs, and DexWild mainly on kinematics (with some video). Humans also rely on touch, sound, and proprioception heavily. Extending our frameworks to multi-modal human data (e.g., wearable tactile sensors on humans, audio of task execution) could inform robots about aspects of manipulation we have not yet utilized (like the feeling of a tight lid, or the sound of a snap-fit).

- **Broadening Beyond Manipulation:** The principle of leveraging human priors can extend to navigation, locomotion, and interaction. For instance, HRP-like strategies could be used to learn from driver dashcam videos for autonomous driving, or from human trajectories in crowds for mobile robot navigation. DexWild-like co-training could apply to learning from human teleoperation of drones or multi-robot systems. Our work lays a foundation that human experiences (both observed and lived) are a rich resource for all kinds of robot learning.

In conclusion, this thesis demonstrated that by **leveraging human priors**, robots can achieve levels of generalization well beyond what is possible with robot data alone. We showed substantial empirical gains and introduced scalable methods to attain them. The future is promising: as more human data becomes available and our algorithms become more sophisticated, we inch closer to robots that can seamlessly operate in our complex, ever-changing world, side by side with humans, learning from us in every way possible.

Bibliography

[1] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu *et al.*, “Ego4d: Around the world in 3,000 hours of egocentric video,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 995–19 012.

[2] D. Shan, J. Geng, M. Shu, and D. Fouhey, “Understanding human hands in contact at internet scale,” in *CVPR*, 2020.

[3] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, “Diffusion policy: Visuomotor policy learning via action diffusion,” in *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[4] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, “What matters in learning from offline human demonstrations for robot manipulation,” in *Conference on Robot Learning (CoRL)*, 2021.

[5] S. Dasari, J. Wang, J. Hong, S. Bahl, Y. Lin, A. S. Wang, A. Thankaraj, K. S. Chahal, B. Calli, S. Gupta *et al.*, “Rb2: Robotic manipulation benchmarking with a twist,” in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[6] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, “Designing network design spaces,” in *CVPR*, 2020.

[7] C. Feichtenhofer, Y. Li, K. He *et al.*, “Masked autoencoders as spatiotemporal learners,” *NeurIPS*, 2022.

[8] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, I. Osband *et al.*, “Deep q-learning from demonstrations,” in *AAAI*, 2018.

[9] A. Kannan, K. Shaw, S. Bahl, P. Mannam, and D. Pathak, “Deft: Dexterous fine-tuning for real-world hand policies,” *CoRL*, 2023.

[10] L. Weihs, U. Jain, I.-J. Liu, J. Salvador, S. Lazebnik, A. Kembhavi, and A. Schwing, "Bridging the imitation gap by adaptive insubordination," *NeurIPS*, 2021.

[11] D. Damen, H. Doughty, G. M. Farinella, , A. Furnari, J. Ma, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100," *IJCV*, 2022.

[12] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.

[13] T. Xiao, I. Radosavovic, T. Darrell, and J. Malik, "Masked visual pre-training for motor control," *arXiv preprint arXiv:2203.06173*, 2022.

[14] S. Dasari, M. K. Srirama, U. Jain, and A. Gupta, "An unbiased look at datasets for visuomotor pre-training," in *Conference on Robot Learning*. PMLR, 2023.

[15] K. Burns, Z. Witzel, J. I. Hamid, T. Yu, C. Finn, and K. Hausman, "What makes pre-trained visual representations successful for robust manipulation?" *ArXiv*, 2023.

[16] M. K. Srirama, S. Dasari, S. Bahl, and A. Gupta, "HRP: Human affordances for robotic pre-training," in *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, 2024.

[17] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.

[18] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real-world robot learning with masked visual pre-training," *CoRL*, 2022.

[19] A. Majumdar, K. Yadav, S. Arnaud, Y. J. Ma, C. Chen, S. Silwal, A. Jain, V.-P. Berges, P. Abbeel, J. Malik *et al.*, "Where are we in the search for an artificial visual cortex for embodied intelligence?" *arXiv preprint arXiv:2303.18240*, 2023.

[20] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1334–1373, 2016.

[21] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 3406–3413.

[22] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press, 2018.

[23] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[24] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 000–16 009.

[25] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, S. Levine, and G. Brain, "Time-contrastive networks: Self-supervised learning from video," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 1134–1141.

[26] S. Schaal *et al.*, "Learning from demonstration," *Advances in neural information processing systems*, pp. 1040–1046, 1997.

[27] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[28] Y. Rong, T. Shiratori, and H. Joo, "Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, October 2021, pp. 1749–1759.

[29] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," *CVPR*, 2021.

[30] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke *et al.*, "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," *arXiv preprint arXiv:1806.10293*, 2018.

[31] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen, "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection," *The International journal of robotics research*, vol. 37, no. 4-5, pp. 421–436, 2018.

[32] A. Gupta, A. Murali, D. P. Gandhi, and L. Pinto, "Robot learning in homes: Improving generalization and reducing dataset bias," *Advances in neural information processing systems*, vol. 31, 2018.

[33] S. Song, A. Zeng, J. Lee, and T. Funkhouser, "Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4978–4985, 2020.

[34] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu,

U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar, P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "Rt-1: Robotics transformer for real-world control at scale," in *arXiv preprint arXiv:2212.06817*, 2022.

[35] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[36] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," *Advances in neural information processing systems*, vol. 1, 1988.

[37] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, "Learning by cheating," in *CoRL*, 2020.

[38] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," in *International Conference on Learning Representations*, 2020.

[39] W. Whitney, R. Agarwal, K. Cho, and A. Gupta, "Dynamics-aware embeddings," *arXiv preprint arXiv:1908.09357*, 2019.

[40] A. V. Nair, V. Pong, M. Dalal, S. Bahl, S. Lin, and S. Levine, "Visual reinforcement learning with imagined goals," *Advances in neural information processing systems*, vol. 31, 2018.

[41] S. Dasari and A. Gupta, "Transformers for one-shot visual imitation," in *Conference on Robot Learning*. PMLR, 2021, pp. 2071–2084.

[42] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and C. Finn, "Robonet: Large-scale multi-robot learning," *arXiv preprint arXiv:1910.11215*, 2019.

[43] O. X.-E. Collaboration, A. Padalkar, A. Pooley, A. Jain, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Singh, A. Brohan, A. Raffin, A. Wahid, B. Burgess-Limerick, B. Kim, B. Schölkopf, B. Ichter, C. Lu, C. Xu, C. Finn, C. Xu, C. Chi, C. Huang, C. Chan, C. Pan, C. Fu, C. Devin, D. Driess, D. Pathak, D. Shah, D. Büchler, D. Kalashnikov, D. Sadigh, E. Johns, F. Ceola, F. Xia, F. Stulp, G. Zhou, G. S. Sukhatme, G. Salhotra, G. Yan, G. Schiavi, H. Su, H.-S. Fang, H. Shi, H. B. Amor, H. I. Christensen, H. Furuta, H. Walke, H. Fang, I. Mordatch, I. Radosavovic, I. Leal, J. Liang, J. Kim, J. Schneider, J. Hsu, J. Bohg, J. Bingham, J. Wu, J. Wu, J. Luo, J. Gu, J. Tan, J. Oh, J. Malik, J. Tompson, J. Yang, J. J. Lim, J. Silvério, J. Han, K. Rao, K. Pertsch, K. Hausman, K. Go, K. Gopalakrishnan, K. Goldberg, K. Byrne, K. Oslund, K. Kawaharazuka, K. Zhang, K. Majd, K. Rana,

K. Srinivasan, L. Y. Chen, L. Pinto, L. Tan, L. Ott, L. Lee, M. Tomizuka, M. Du, M. Ahn, M. Zhang, M. Ding, M. K. Srirama, M. Sharma, M. J. Kim, N. Kanazawa, N. Hansen, N. Heess, N. J. Joshi, N. Suenderhauf, N. D. Palo, N. M. M. Shafiullah, O. Mees, O. Kroemer, P. R. Sanketi, P. Wohlhart, P. Xu, P. Sermanet, P. Sundaresan, Q. Vuong, R. Rafailov, R. Tian, R. Doshi, R. Martín-Martín, R. Mendonca, R. Shah, R. Hoque, R. Julian, S. Bustamante, S. Kirmani, S. Levine, S. Moore, S. Bahl, S. Dass, S. Song, S. Xu, S. Haldar, S. Adebola, S. Guist, S. Nasiriany, S. Schaal, S. Welker, S. Tian, S. Dasari, S. Belkhale, T. Osa, T. Harada, T. Matsushima, T. Xiao, T. Yu, T. Ding, T. Davchev, T. Z. Zhao, T. Armstrong, T. Darrell, V. Jain, V. Vanhoucke, W. Zhan, W. Zhou, W. Burgard, X. Chen, X. Wang, X. Zhu, X. Li, Y. Lu, Y. Chebotar, Y. Zhou, Y. Zhu, Y. Xu, Y. Wang, Y. Bisk, Y. Cho, Y. Lee, Y. Cui, Y. hua Wu, Y. Tang, Y. Zhu, Y. Li, Y. Iwasawa, Y. Matsuo, Z. Xu, and Z. J. Cui, “Open X-Embodiment: Robotic learning datasets and RT-X models,” 2024.

[44] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, “Vip: Towards universal visual reward and representation via value-implicit pre-training,” *arXiv preprint arXiv:2210.00030*, 2022.

[45] J. Wang, S. Dasari, M. K. Srirama, S. Tulsiani, and A. Gupta, “Manipulate by seeing: Creating manipulation controllers from pre-trained representations,” 2023.

[46] K. Shaw, S. Bahl, and D. Pathak, “Videodex: Learning dexterity from internet videos,” in *CoRL*, 2022.

[47] P. Mandikal and K. Grauman, “Dexvip: Learning dexterous grasping with human hand pose priors from video,” in *Conference on Robot Learning*. PMLR, 2022, pp. 651–661.

[48] J. Gibson, “The ecological approach to visual perception,” *Houghton Mifflin Comp*, 1979.

[49] J. J. Gibson, *The senses considered as perceptual systems*, vol. 2, no. 1.

[50] D. Eigen and R. Fergus, “Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. corr, abs/1411.4734,” *arXiv preprint arXiv:1411.4734*, 2014.

[51] A. Bansal, B. Russell, and A. Gupta, “Marr revisited: 2d-3d alignment via surface normal prediction,” in *CVPR*, 2016.

[52] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert, “From 3d scene geometry to human workspace,” in *CVPR*, 2011.

[53] Y. Zhu, C. Jiang, Y. Zhao, D. Terzopoulos, and S.-C. Zhu, “Inferring forces and learning human utilities from videos,” in *CVPR*, 2016.

[54] M. Hassanin, S. Khan, and M. Tahtali, “Visual affordance and function understanding: a survey. arxiv,” *arXiv preprint arXiv:1807.06775*, 2018.

[55] Y. Zhao and S.-C. Zhu, “Scene parsing by integrating function, geometry and appearance models,” in *CVPR*, 2013.

[56] A. Myers, C. L. Teo, C. Fermüller, and Y. Aloimonos, “Affordance detection of tool parts from geometric features,” in *ICRA)*, 2015.

[57] A. Roy and S. Todorovic, “A multi-scale cnn for affordance segmentation in rgb images,” in *ECCV*, 2016.

[58] J. Sawatzky, A. Srikantha, and J. Gall, “Weakly supervised affordance detection,” in *CVPR*, 2017.

[59] H. S. Koppula and A. Saxena, “Anticipating human activities using object affordances for reactive robotic response,” *TPAMI*, 2015.

[60] N. Rhinehart and K. M. Kitani, “Learning action maps of large environments via first-person vision,” in *CVPR*, 2016.

[61] J. Gao, Z. Yang, and R. Nevatia, “Red: Reinforced encoder-decoder networks for action anticipation,” *arXiv preprint arXiv:1707.04818*, 2017.

[62] A. Jain, A. Singh, H. S. Koppula, S. Soh, and A. Saxena, “Recurrent neural networks for driver activity anticipation via sensory-fusion architecture,” in *ICRA*, 2016.

[63] D.-A. Huang and K. M. Kitani, “Action-reaction: Forecasting the dynamics of human interaction,” in *ECCV*, 2014.

[64] C. Vondrick, D. Oktay, H. Pirsiavash, and A. Torralba, “Predicting motivations of actions by leveraging text,” in *CVPR*, 2016.

[65] Y. Abu Farha, A. Richard, and J. Gall, “When will you do what?-anticipating temporal occurrences of activities,” in *CVPR*, 2018.

[66] T. Lan, T.-C. Chen, and S. Savarese, “A hierarchical representation for future action prediction,” in *ECCV*, 2014.

[67] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee, "Decomposing motion and content for natural video sequence prediction," *arXiv preprint arXiv:1706.08033*, 2017.

[68] A. Furnari and G. M. Farinella, "Rolling-unrolling lstms for action anticipation from first-person video," *TPAMI*, 2020.

[69] E. V. Mascaro, H. Ahn, and D. Lee, "Intention-conditioned long-term human egocentric action forecasting@ ego4d challenge 2022," *arXiv preprint arXiv:2207.12080*, 2022.

[70] R. Girdhar and K. Grauman, "Anticipative video transformer," in *ICCV*, 2021.

[71] Y. Zhu, A. Fathi, and L. Fei-Fei, "Reasoning about object affordances in a knowledge base representation," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*. Springer, 2014, pp. 408–424.

[72] A. Furnari, S. Battiato, K. Grauman, and G. M. Farinella, "Next-active-object prediction from egocentric videos," *Journal of Visual Communication and Image Representation*, 2017.

[73] M. Goyal, S. Modi, R. Goyal, and S. Gupta, "Human hands as probes for interactive object understanding," in *CVPR*, 2022.

[74] T. Nagarajan, C. Feichtenhofer, and K. Grauman, "Grounded human-object interaction hotspots from video," in *ICCV*, 2019.

[75] S. Liu, S. Tripathi, S. Majumdar, and X. Wang, "Joint hand motion and interaction hotspots prediction from egocentric videos," in *CVPR*, 2022.

[76] Y. Ye, X. Li, A. Gupta, S. D. Mello, S. Birchfield, J. Song, S. Tulsiani, and S. Liu, "Affordance diffusion: Synthesizing hand-object interactions," in *CVPR*, 2023.

[77] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Scaling egocentric vision: The epic-kitchens dataset," in *European Conference on Computer Vision (ECCV)*, 2018.

[78] P. Das, C. Xu, R. F. Doell, and J. J. Corso, "A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2634–2641.

[79] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Scaling egocentric vision: The epic-kitchens dataset," in *ECCV*, 2018.

[80] Y. Liu, Y. Liu, C. Jiang, K. Lyu, W. Wan, H. Shen, B. Liang, Z. Fu, H. Wang, and L. Yi, "Hoi4d: A 4d egocentric dataset for category-level human-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21 013–21 022.

[81] A. Darkhalil, D. Shan, B. Zhu, J. Ma, A. Kar, R. Higgins, S. Fidler, D. Fouhey, and D. Damen, "Epic-kitchens visor benchmark: Video segmentations and object relations," *Advances in Neural Information Processing Systems*, vol. 35, pp. 13 745–13 758, 2022.

[82] S. Bahl, R. Mendonca, L. Chen, U. Jain, and D. Pathak, "Affordances from human videos as a versatile representation for robotics," 2023.

[83] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," *CoRR*, vol. abs/1712.06584, 2017. [Online]. Available: http://arxiv.org/abs/1712.06584

[84] J. Wang, F. Mueller, F. Bernard, S. Sorli, O. Sotnychenko, N. Qian, M. A. Otaduy, D. Casas, and C. Theobalt, "Rgb2hands: real-time tracking of 3d hand interactions from monocular rgb video," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, pp. 1–16, 2020.

[85] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv preprint arXiv:1906.08172*, 2019.

[86] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, "Scaling vision transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 12 104–12 113.

[87] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016.

[88] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9729–9738.

[89] H. Zhou, Y. Yuan, and C. Shi, "Object tracking using sift features and mean shift," *Computer vision and image understanding*, vol. 113, no. 3, pp. 345–352, 2009.

[90] R. M. Shah and V. Kumar, "Rrl: Resnet as representation for reinforcement learning," in *ICML*, 2021.

[91] J. Pari, N. Muhammad, S. P. Arunachalam, L. Pinto *et al.*, "The surprising effectiveness of representation learning for visual imitation," *arXiv preprint arXiv:2112.01511*, 2021.

[92] W. Zhou, B. Jiang, F. Yang, C. Paxton, and D. Held, "Hacman: Learning hybrid actor-critic maps for 6d non-prehensile manipulation," in *7th Annual Conference on Robot Learning*, 2023.

[93] Y. Ju, K. Hu, G. Zhang, G. Zhang, M. Jiang, and H. Xu, "Robo-abc: Affordance generalization beyond categories via semantic correspondence for robot manipulation," *arXiv preprint arXiv:2401.07487*, 2024.

[94] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in *CoRL*, 2022.

[95] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, and J. Lee, "Transporter networks: Rearranging the visual world for robotic manipulation," *CoRL*, 2020.

[96] M. Chang, A. Prakash, and S. Gupta, "Look ma, no hands! agent-environment factorization of egocentric videos," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[97] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "Voxposer: Composable 3d value maps for robotic manipulation with language models," *arXiv preprint arXiv:2307.05973*, 2023.

[98] H. Bharadhwaj, A. Gupta, V. Kumar, and S. Tulsiani, "Towards generalizable zero-shot manipulation via translating human interaction plans," *arXiv preprint arXiv:2312.00775*, 2023.

[99] A. Giannou, S. Rajput, and D. Papailiopoulos, "The expressive power of tuning only the normalization layers," *arXiv preprint arXiv:2302.07937*, 2023.

[100] B. Zhao, H. Tu, C. Wei, J. Mei, and C. Xie, "Tuning layernorm in attention: Towards efficient multi-modal llm finetuning," *arXiv preprint arXiv:2312.11420*, 2023.

[101] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[102] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, vol. abs/2103.00020, 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[103] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[104] S. Ross, G. Gordon, and D. Bagnell, “A reduction of imitation learning and structured prediction to no-regret online learning,” in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[105] X. Zhou, R. Girdhar, A. Joulin, P. Krähenbühl, and I. Misra, “Detecting twenty-thousand classes using image-level supervision,” *arXiv preprint arXiv:2201.02605*, 2022.

[106] K. Shaw, A. Agarwal, and D. Pathak, “Leap hand:low-cost, efficient, and anthropomorphic hand for robot learning,” *RSS*, 2023.

[107] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[108] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *NeurIPS*, 2017.

[109] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.

[110] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language models are few-shot learners,” 2020.

[111] H. Liu, C. Li, Q. Wu, and Y. J. Lee, “Visual instruction tuning,” in *NeurIPS*, 2023.

[112] A. Steiner, A. S. Pinto, M. Tschannen, D. Keysers, X. Wang, Y. Bitton, A. Gritsenko, M. Minderer, A. Sherbondy, S. Long, S. Qin, R. Ingle, E. Bugliarello, S. Kazemzadeh, T. Mesnard, I. Alabdulmohsin, L. Beyer, and X. Zhai, “PaliGemma 2: A Family of Versatile VLMs for Transfer,” *arXiv preprint arXiv:2412.03555*, 2024.

[113] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis *et al.*, “Droid: A large-scale in-the-wild robot manipulation dataset,” *arXiv preprint arXiv:2403.12945*, 2024.

[114] H. Walke, K. Black, A. Lee, M. J. Kim, M. Du, C. Zheng, T. Zhao, P. Hansen-Estruch, Q. Vuong, A. He, V. Myers, K. Fang, C. Finn, and S. Levine, “Bridgedata v2: A dataset for robot learning at scale,” in *Conference on Robot Learning (CoRL)*, 2023.

[115] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Epic-kitchens: A large-scale dataset for recognizing, anticipating, and retrieving hand-object interactions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 802–819.

[116] Y. Hu, Q. Xie, V. Jain, J. Francis, J. Patrikar, N. Keetha, S. Kim, Y. Xie, T. Zhang, Z. Zhao *et al.*, "Toward general-purpose robots via foundation models: A survey and meta-analysis," *arXiv preprint arXiv:2312.08782*, 2023.

[117] K. Shaw, S. Bahl, and D. Pathak, "Videodex: Learning dexterity from internet videos," in *Conference on Robot Learning*. PMLR, 2023, pp. 654–665.

[118] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[119] C. Wang, H. Shi, W. Wang, R. Zhang, L. Fei-Fei, and K. Liu, "Dexcap: Scalable and portable mocap data collection system for dexterous manipulation," in *Robotics: Science and Systems (RSS)*, 2024.

[120] K. Shaw, S. Bahl, and D. Pathak, "Videodex: Learning dexterity from internet videos," in *Proceedings of The 6th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, K. Liu, D. Kulic, and J. Ichnowski, Eds., vol. 205. PMLR, 14–18 Dec 2023, pp. 654–665.

[121] O. Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo *et al.*, "Octo: An open-source generalist robot policy," *Proceedings of Robotics: Science and Systems, Delft, Netherlands*, 2023.

[122] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.

[123] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, "What matters in learning from offline human demonstrations for robot manipulation," *arXiv preprint arXiv:2108.03298*, 2021.

[124] H. G. Singh, A. Loquercio, C. Sferrazza, J. Wu, H. Qi, P. Abbeel, and J. Malik, "Hand-object interaction pretraining from videos," 2024. [Online]. Available: https://arxiv.org/abs/2409.08273

[125] S. Ye, J. Jang, B. Jeon, S. Joo, J. Yang, B. Peng, A. Mandlekar, R. Tan, Y.-W. Chao, B. Y. Lin *et al.*, "Latent action pretraining from videos," *arXiv preprint arXiv:2410.11758*, 2024.

[126] R. Singh, A. Allshire, A. Handa, N. Ratliff, and K. Van Wyk, "Dextrah-rgb: Visuomotor policies to grasp anything with dexterous hands," *arXiv preprint arXiv:2412.01791*, 2024.

[127] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," 2023.

[128] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, "Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators," 2023.

[129] K. Shaw, Y. Li, J. Yang, M. K. Srirama, R. Liu, H. Xiong, R. Mendonca, and D. Pathak, "Bimanual dexterity for complex tasks," in *8th Annual Conference on Robot Learning*, 2024.

[130] A. Iyer, Z. Peng, Y. Dai, I. Guzey, S. Haldar, S. Chintala, and L. Pinto, "OPEN TEACH: A versatile teleoperation system for robotic manipulation," *arXiv preprint arXiv:2403.07870*, 2024.

[131] H. Etukuru, N. Naka, Z. Hu, S. Lee, J. Mehu, A. Edsinger, C. Paxton, S. Chintala, L. Pinto, and N. M. M. Shafiullah, "Robot utility models: General policies for zero-shot deployment in new environments," 2024.

[132] F. Lin, Y. Hu, P. Sheng, C. Wen, J. You, and Y. Gao, "Data scaling laws in imitation learning for robotic manipulation," in *Conference on Robot Learning (CoRL)*, 2024.

[133] G. Pavlakos, D. Shan, I. Radosavovic, A. Kanazawa, D. Fouhey, and J. Malik, "Reconstructing hands in 3d with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[134] Y. Rong, T. Shiratori, and H. Joo, "Frankmocap: A monocular 3d whole-body pose estimation system via regression and integration," in *ICCV Workshops*, 2021.

[135] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang, "Open-television: Teleoperation with immersive active visual feedback," *arXiv preprint arXiv:2407.01512*, 2024.

[136] A. Sivakumar, K. Shaw, and D. Pathak, "Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube," *RSS*, 2022.

[137] G. Pavlakos, D. Shan, I. Radosavovic, A. Kanazawa, D. Fouhey, and J. Malik, "Hamer: Hand mesh recovery for the egoexo4d hand pose challenge."

[138] S. Kareer, D. Patel, R. Punamiya, P. Mathur, S. Cheng, C. Wang, J. Hoffman, and D. Xu, "Egomimic: Scaling imitation learning via egocentric video," 2024.

[139] R.-Z. Qiu, S. Yang, X. Cheng, C. Chawla, J. Li, T. He, G. Yan, D. J. Yoon, R. Hoque, L. Paulsen, G. Yang, J. Zhang, S. Yi, G. Shi, and X. Wang, "Humanoid policy ˜ human policy," *arXiv preprint arXiv:2503.13441*, 2025.

[140] J. A. Corrales, F. A. Candelas, and F. Torres, "Hybrid tracking of human operators using imu/uwb data fusion by a kalman filter," in *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction*, 2008, pp. 193–200.

[141] Y. Tian, X. Meng, D. Tao, D. Liu, and C. Feng, "Upper limb motion tracking with the integration of imu and kinect," *Neurocomputing*, vol. 159, pp. 207–218, 2015.

[142] A. Pfister, A. M. West, S. Bronner, and J. A. Noah, "Comparative abilities of microsoft kinect and vicon 3d motion capture for gait analysis," *Journal of medical engineering & technology*, vol. 38, no. 5, pp. 274–280, 2014.

[143] H. Zhang, S. Hu, Z. Yuan, and H. Xu, "Doglove: Dexterous manipulation with a low-cost open-source haptic force feedback glove," *arXiv preprint arXiv:2502.07730*, 2025.

[144] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox, "Freihand: A dataset for markerless capture of hand pose and shape from single rgb images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 813–822.

[145] Z. Fan, O. Taheri, D. Tzionas, M. Kocabas, M. Kaufmann, M. J. Black, and O. Hilliges, "Arctic: A dataset for dexterous bimanual hand-object manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 943–12 954.

[146] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield *et al.*, "Dexycb: A benchmark for capturing hand grasping of objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9044–9053.

[147] Valve Corporation, https://store.steampowered.com/steamvr, [Virtual reality platform].

[148] A. from UC San Diego and MIT, "Open-television: An open-source immersive teleoperation system with stereo visual feedback," *The Robot Report*, 2024.

[149] A. Handa, K. Van Wyk, W. Yang, J. Liang, Y.-W. Chao, Q. Wan, S. Birchfield, N. Ratliff, and D. Fox, "Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9164–9170.

[150] A. Sivakumar, K. Shaw, and D. Pathak, "Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube," 2022.

[151] H. Ha, Y. Gao, Z. Fu, J. Tan, and S. Song, "UMI on legs: Making manipulation policies mobile with manipulation-centric whole-body controllers," in *Proceedings of the 2024 Conference on Robot Learning*, 2024.

[152] I. Radosavovic, T. Xiao, S. James, P. Abbeel, J. Malik, and T. Darrell, "Real-world robot learning with masked visual pre-training," *CoRL*, 2022.