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

MotionTrans: Human VR Data Enable Motion-Level Learning for Robotic Manipulation Policies

Chengbo Yuan[1,2], Rui Zhou*[5], Mengzhen Liu*[3], Yingdong Hu[1,2], Shengjie Wang[1,2]
Li Yi[1,2], Chuan Wen[4], Shanghang Zhang[3], Yang Gao[1,2]†
[1]Institute for Interdisciplinary Information Sciences, Tsinghua University [2]Shanghai Qi Zhi Institute
[3]State Key Laboratory of Multimedia Information Processing, School of Computer Science, Peking University
[4]Shanghai Jiao Tong University [5]Wuhan University
* Indicates equal contribution. † The corresponding author.

https://motiontrans.github.io/

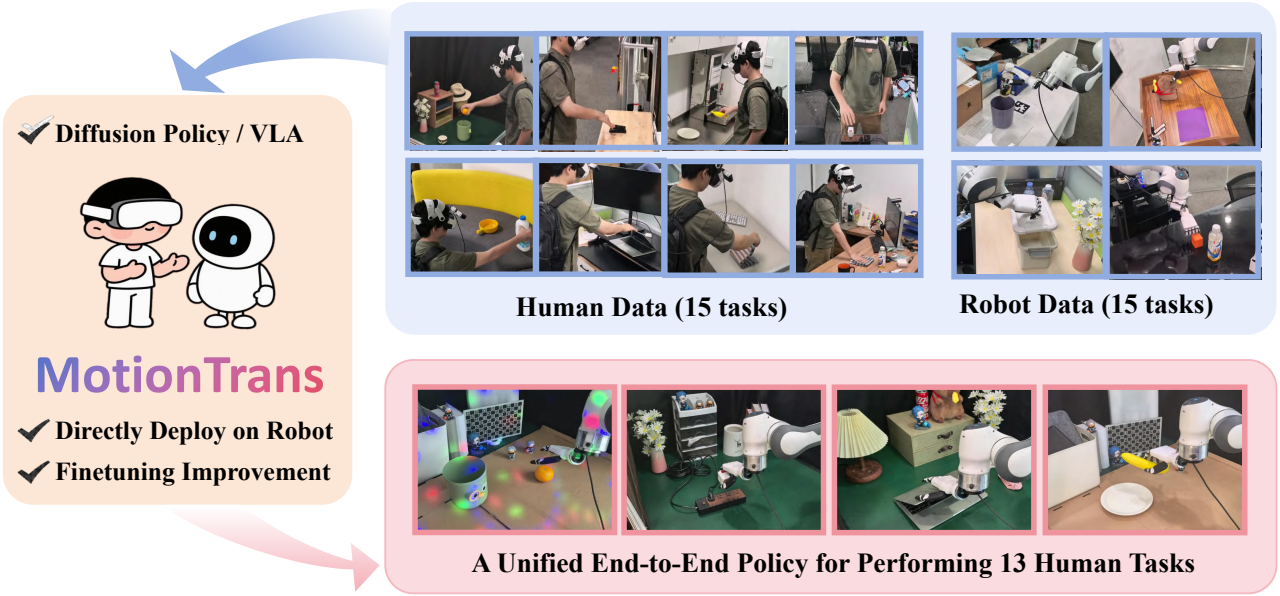

Fig. 1: We propose *MotionTrans*, a framework for **motion-level** learning from VR human data. By cotraining on 15 human tasks and 15 robot tasks, we enable end-to-end manipulation policies to directly perform tasks present in human data on real robots. When a few robot demonstrations are available, finetuning performance further improves.

***Abstract*— Scaling real robot data is a key bottleneck in imitation learning, leading to the use of auxiliary data for policy training. While other aspects of robotic manipulation such as image or language understanding may be learned from internet-based datasets, acquiring motion knowledge remains challenging. Human data, with its rich diversity of manipulation behaviors, offers a valuable resource for this purpose. While previous works show that using human data can bring benefits, such as improving robustness and training efficiency, it remains unclear whether it can realize its greatest advantage: *enabling robot policies to directly learn new motions for task completion*. In this paper, we systematically explore this potential through multi-task human-robot cotraining. We introduce *MotionTrans*, a framework that includes a data collection system, a human data transformation pipeline, and a weighted cotraining strategy. By cotraining 30 human-robot tasks simultaneously, we direcly transfer motions of 13 tasks from human data to deployable end-to-end robot policies. Notably, 9 tasks achieve non-trivial success rates in zero-shot manner. *MotionTrans* also significantly enhances pretraining-finetuning performance (+40% success rate). These findings unlock the potential of motion-level learning from human data, offering insights into its effective use for training robotic manipulation policies. All data, code, and model weights will be open-sourced.**

I. Introduction

Learning robotic manipulation policies from teleoperated demonstrations has advanced rapidly in recent years [1]–[3]. However, collecting large-scale robot datasets is costly and labor-intensive [4], creating a major bottleneck for scaling manipulation capabilities. To mitigate data scarcity, researchers have turned to auxiliary sources such as images and language [5] to train policies. While internet data provides abundant vision–language priors [5], acquiring motion knowledge remains challenging.

Human data [6], [7] is abundant, easy to collect, and rich in diverse manipulation behaviors [7], making it a particularly promising source. Prior work leverages human demonstrations to extract task-aware intermediates—e.g., affordances [8] and keypoint flows [9]—to support motion transfer, but introducing intermediate representations hinders

seamless integration with mainstream end-to-end policies. More recently, advances in wearable sensing have enabled direct use of human motion data (e.g., VR-tracked hand poses) for robot policy pretraining or cotraining [6], [10]–[13], yielding gains in visual grounding [12], robustness [11], and data efficiency [13]. Yet, it remains unclear whether such data can deliver its greatest advantage: *enabling robot policies to directly acquire new task motions*.

We address this question with *MotionTrans*, a framework designed to **directly learn 10+ robot-executable motions from human data within a unified, end-to-end policy**. We achieve this via multi-task human–robot cotraining. Concretely, we build a VR-based teleoperation system and data pipeline to construct the *MotionTrans Dataset*, comprising 3,213 demonstrations across 15 human tasks and 15 robot tasks in over 10 scenes. We introduce a transformation procedure that maps human demonstrations into the robot observation–action space, making them compatible with mainstream end-to-end policies such as Diffusion Policy [2] and the Vision–Language–Action model (π_0-VLA) [3]. Finally, we adopt a weighted cotraining strategy that jointly optimizes over human and robot tasks. We name the framework *MotionTrans* because it transfers human motions into deployable robot policies.

We first evaluate zero-shot performance on all human tasks, directly deploying policies to the robot without collecting any robot data for those tasks. Results show that both Diffusion Policy [2] and π_0-VLA [3] achieve non-trivial success on 9 tasks. Even when unsuccessful, the policies often exhibit meaningful task-directed motions, such as reaching target objects. In the few-shot setting, pretraining on the *MotionTrans Dataset* yields an average 40% boost in success rate when a small number of robot demonstrations are available. Together, these findings demonstrate the feasibility of motion-level learning from human data and provide a practical framework and principles for doing so. Our contributions can be summarized as:

- *MotionTrans*, a framework for end-to-end human-to-robot motion transfer, including data collection system, *MotionTrans Dataset*, a pipeline to transform human data into robot format, and a human-robot cotraining strategy.
- ***MotionTrans* enables explicit human motions transfer for end-to-end robot policies, even for zero-shot settings** (directly learn 13 tasks from human data). It also improve finetuning performance with +40% success rate on average.

II. Related Work

A. Imitation Learning for Robot Manipulation

Imitation learning [14], [15] has made significant progress in recent years. By learning motion from training data [1], [16], imitation policies can effectively perform a wide range of manipulation tasks [2], including challenging multi-task settings [3], [13]. In this paper, we focus on two widely-used architectures for imitation learning: Diffusion Policy [2] and the π_0 Vision-Language-Action Model (π_0-VLA) [3]. However, the scalability of training data remains a major challenge, due to the high cost of collecting real-robot data [4]. This has led to the use of auxiliary data [17], [18] for policy training. Despite ability such as image or language understanding in robotic manipulation could improve from internet-based pretraining [5], [19], acquiring motion knowledge remains difficult. Human data [7], [20], with its abundant and diverse manipulation behaviors, provides a valuable supplement.

B. Task-Aware Representation Learning from Human

Early works have leveraged task-aware representations for human-to-robot knowledge transfer. Self-supervised learning has been used for implicit task-aware representations [21], [22] learning, while representations like affordances [8], object poses [23], videos [24], and motion flows [9], [25] support motion-aware representation learning. EgoZero [26] predicts wrist poses from smart glasses, but relies on keypoint-based representations for policy observations. The use of intermediate representations in these methods limits their integration with mainstream end-to-end visuomotor policy learning [2], [3], restricting their future applicability.

C. End-to-End Policy Learning with Posed Human Data

Recent advancements in wearable sensing [6], [16] now allow easy collection of posed human data (with hand keypoints, wrist poses information etc.) through VR devices [7]. This data provide action label for prediction, supporting end-to-end policy learning [27]. Some studies cotrain human and robot data [6], [10], [28], [29], while others first pretrain with human data and then finetune with robot demonstrations [11]–[13]. These works have shown policy improvements in visual grounding [12], robustness [6], [11], and training efficiency [10], [13]. However, whether it can achieve direct transfer of motions from human to robot remains unclear [26]. To the best of our knowledge, our paper is the first to systematically verify motion-level end-to-end learning from human data.

III. MotionTrans

In this section, we present the *MotionTrans* framework (Figure 2). We first formalize the motion transfer problem and define the policy's observation–action space (Section III-A). To enable human–robot cotraining, we then describe the data collection systems for both human and robot data (Section III-B). Next, we propose a pipeline that converts human data into robot format (Section III-C), ensuring compatibility with mainstream end-to-end robot policies training. Finally, we choose the architecture of robot policies and apply human-robot multi-task cotraining (Section III-D).

A. Problem Definition

Our goal is to enable explicit human-to-robot motion transfer. Considering the embodiment gap between human and robot [10], we explore this problem within a multi-task human–robot cotraining framework, where robot data for certain tasks are available to help motions in human data adapt to the robot. We train a policy P_{policy} on $D = D_{\text{robot}} \cup D_{\text{human}}$, where $D_{\text{robot}} = \{D^i_{\text{robot}} \mid i = 1, \ldots, N_{\text{robot}}\}$ and

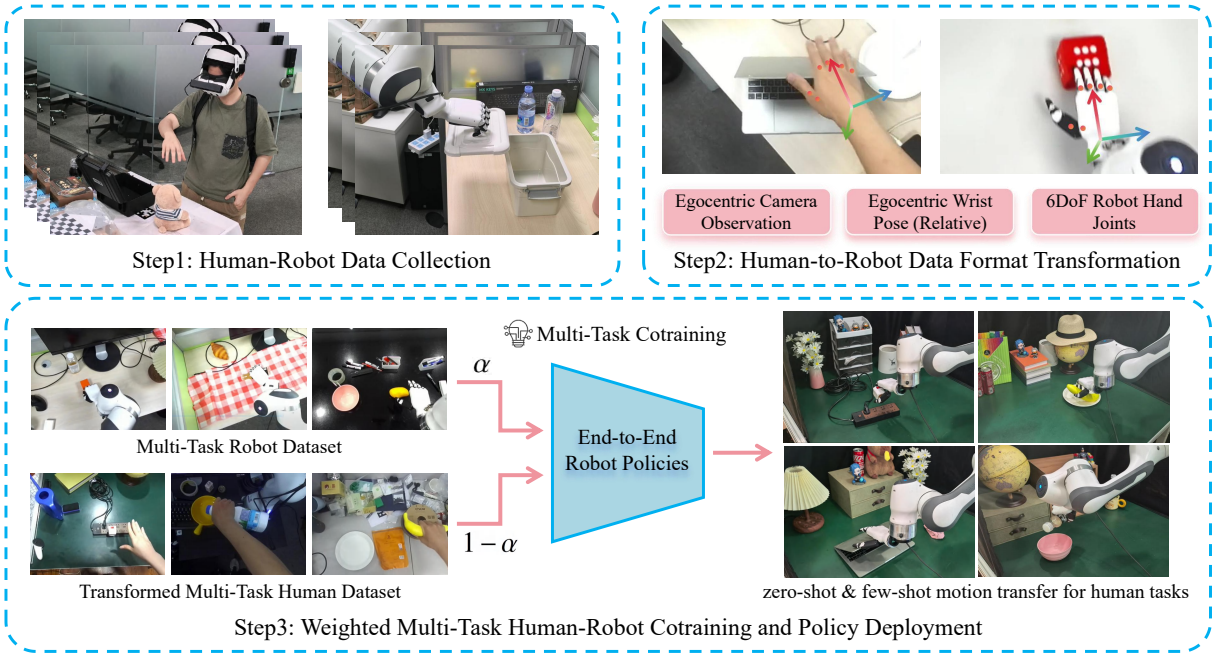

Fig. 2: Overview of *MotionTrans*: a human–robot data collection system, a pipeline that transforms human data into a robot-compatible format, and a weighted human–robot multi-task cotraining strategy. The trained policies can be directly deployed to execute tasks appearing only in the human dataset on real robots.

$D_{\text{human}} = \{D^i_{\text{human}} \mid i = 1, \ldots, N_{\text{human}}\}$. Each D^i denotes a task-specific subset, and the human and robot task sets are **non-overlapping**. After training, we deploy P_{policy} on robot and evaluate it on **tasks from D_{human}** to assess the effectiveness of motion transfer. This constitutes the **zero-shot** setting, as these evaluated human tasks have no robot demonstrations during training. We also consider a **few-shot finetuning** setting, where a small number of robot demonstrations for tasks in D_{human} are used to finetune P_{policy}.

We define the input and output of our policies within the robot observation-action space $S = (I_t, P_t, A_t)$. At each timestamp t, the policy receives an egocentric RGB image $I_t \in \mathbb{R}^{H\times W\times 3}$ and proprioceptive states $P_t \in \mathbb{R}^{T_P\times D}$, where T_P is the history length and D is the state dimension. For simplicity, this work focuses on single-arm tasks (Figure 3), thus D corresponds to the concatenation of one robot wrist pose and one robot hand joint state (Figure 4(c)). The policy outputs an action chunk prediction $A_t \in \mathbb{R}^{T_A\times D}$ [2], where T_A denotes the action prediction horizon. In this paper, we set $T_p = 2$ and $T_A = 16$.

B. Human-Robot Data Collection System

For human-robot cotraining, we need to collect both robot and human data [10]. Here we describe the data collection system, illustrated in the top-left of Figure 2.

Portable VR-based Human Data Collection. We extend ARCap [16] to build our human data collection system (Figure 4(a)), incorporating a portable VR headset for recording hand keypoint positions K_t, wrist poses W_t and camera pose, and an RGB camera for the image stream I_t. Both the hand keypoints and wrist poses are expressed in the RGB camera coordinate frame. Operators are instructed to minimize head motion to approximate the static camera configuration of typical robot hardware, while allowing slight movements [6]. The view of collectors are provided in Figure 4(b).

Robot Data Collection via Teleoperation. To enable direct human-to-robot motion transfer, the robot hardware must match the functionality of the human arm and hand. Therefore, We use a single robot arm paired with a robot hand (Figure 4(c)). We develop our teleoperation system on OpenTelevision [1], which captures human wrist and hand poses in real time via a VR device and drives the robot to replicate these motions.

C. Transforming Human Data to the Robot Format

Raw human demonstrations collected with VR differ in format from robot demonstrations, preventing direct cotraining with robot policies [11], [12]. We therefore first transform human data into the **robot observation–action space** [1]. After transformation, human data could acts as "supplementary robot data", thus can be used to train any mainstream end-to-end **robot** policy.

Transforming Observation-Action Space. The observation-action space of the robot includes three components: image observation I_t, proprioceptive state P_t, and action A_t. Both P_t and A_t are generated by stacking wrist poses W_t and hand joint states H_t. Next, we describe the design for these components: **(1) Image observation I_t:** We use **egocentric** view for both human and robot data, as shown in Figure 3. **(2) Wrist poses W_t:** We use the **egocentric** camera coordinate system (camera captures I_t) for both human and robot data. This allows for the measurement of wrist poses in a unified coordinate system, ensuring that the spatial definitions of human and robot data are consistent. **(3) Hand joints state H_t:** we employ the

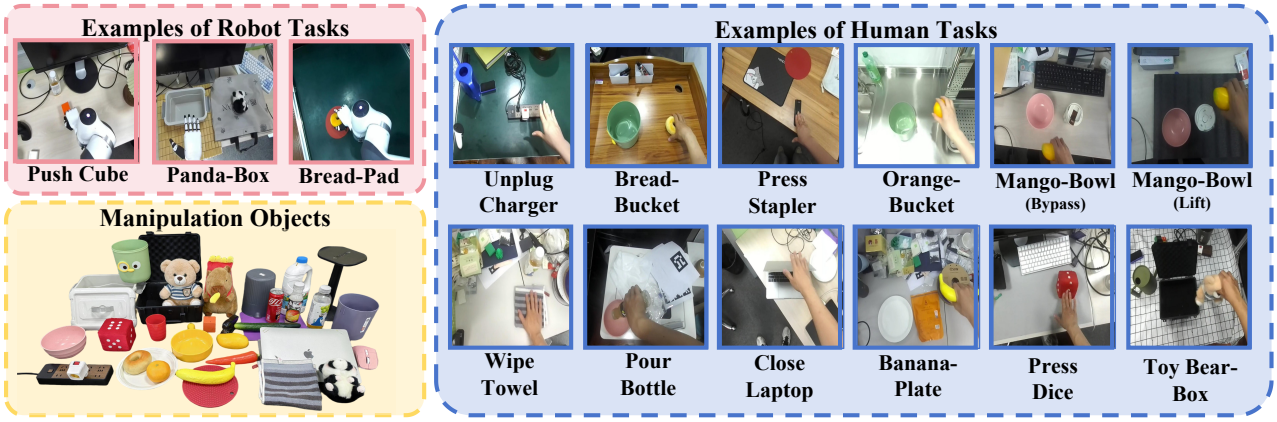

Fig. 3: The *MotionTrans Dataset* comprises 3,213 demonstrations spanning 15 human tasks and 15 robot tasks across more than 10 scenes. For statistical analysis, tasks are grouped by motion-similar skill categories. For human task "Open Box+Panda-Box", it contains both open and pick-place skills.

dex-retargeting [30], an optimization-based inverse kinematics solver, to map human hand keypoints K_t to robot hand joint state H_t.

To further mitigate the difference between human and robot data: (1) we slow down human data by a factor of 2.25 via poses and hand joints state interpolation; (2) we utilize action-chunk-based relative poses [2], [31] for wrist poses to reduce distribution mismatches between human and robot data. For instance, even if the robot's and human's hand positions differ in world space, their relative poses remain the same if they move forward at the same speed; (3) we encourage collectors to change viewpoints between trajectory recordings. This enhances the diversity of positional relationships between the camera view and the manipulation objects, thereby encouraging policies to adapt to a larger distribution of hand poses.

D. Weighted Multi-Task Human–Robot cotraining

By unifying the observation and action spaces, we enable joint training of human and robot data under a shared end-to-end robot policy. This section introduce the multi-task policy architectures we use and how we train these policies.

End-to-End Multi-Task Policy Architectures. We evaluate two representative policy architectures: (1) **Diffusion Policy (DP)** [2]: we extend DP from single-task to multi-task by associating each task with a learnable embedding used as a task condition. We also replace the visual encoder with DINOv2 [32] to strengthen perception [14]. (2) **Vision–Language–Action model (π_0-VLA)** [3]: a policy that integrates large-scale pretrained vision–language models [33] for multimodal perception and instruction following. We load π_0-droid pretrained checkpoints [34] before training. Since π_0-VLA supports language input, we specify tasks via natural-language instructions.

Weighted Human–Robot Cotraining. Human and robot datasets are often imbalanced [6], [29]. Following [35], we adopt a size-aware weighted objective over $D = D_{\text{robot}} \cup D_{\text{human}}$: $\mathcal{L}_D = \alpha \mathcal{L}_{D_{\text{robot}}} + (1-\alpha)\mathcal{L}_{D_{\text{human}}}$, where $\mathcal{L}$ denotes the loss function of imitation learning [2], [3]. We set $\alpha = \frac{|D_{\text{human}}|}{|D_{\text{human}}|+|D_{\text{robot}}|}$, where $|D_{\text{robot}}|$ and $|D_{\text{human}}|$ denote dataset sizes. This choice compensates for size differences between sources, promoting balanced contributions during training.

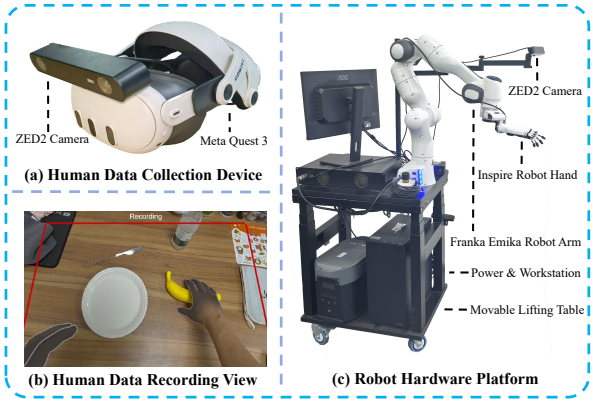

Fig. 4: Illustration of our hardware system, which includes a human VR-based data collection device and a single-arm robot platform. A screenshot of the VR device during human data collection is also provided.

IV. Experiments

In this section, we conduct experiments to verify the effectiveness of *MotionTrans* for human-to-robot motion transfer. We first introduce our detailed experiment setup in Section IV-A, including human-robot hardware platform, training datasets and evaluation tasks and metric. We then conduct experiments for both zero-shot (Section IV-B) and few-shot (Section IV-C) settings, as demonstrated in Section III-A.

A. Experiment Setup

Hardware Platform. For the robot hardware (Figure 4(c)), we use a Franka Emika robot arm in combination with a 6DoF Inspired Dexterous (Right) Hand [1]. This combination mimics the functionality of a human right hand and arm. The robot is mounted on a movable lift table to facilitate data collection in various locations. A ZED2 camera is fixed to the table in an egocentric view to provide an image observation stream. The recorded images are cropped to 640×480 resolution.

For human data collection (Figure 4(a)), we use the Meta Quest 3 as our VR headset. To ensure consistency in image observations, we also employ a ZED2 camera to record RGB images and perform image cropping, using the same setup as in the robot hardware platform [10]. The camera is attached to VR headset by a 3D-printed mounter [16].

***MotionTrans* Multi-Task Dataset.** Here we introduce the *MotionTrans Dataset*, which is used to train our policies. The dataset contains 3,213 demonstrations across more than 10 scenes, covering 15 human tasks and 15 robot tasks. A brief summary of the dataset is shown in Figure 3. The complete task list and the visualizations for all 30 tasks are provided in *supplementary video*. The number of demonstrations for each human / robot task ranges from 40 to 150. To enrich language instructions for VLA training, we leverage GPT-4o [36] to paraphrase and expand task descriptions in the dataset.

For tasks, the human and robot task sets are non-overlapping. For motions, similar tasks across human and robot data (e.g., pick-and-place) share similar motion patterns but still exhibit notable differences. In addition, some motions appear only in the human dataset but not in the robot dataset, such as unplugging, closing, lifting, etc. Overall, the dataset covers a wide range of motions and skills, including pick-and-place, pouring, wiping, pushing, opening, etc. For simplicity, we name pick-place task with "pick object-place target" format, and name other task with "verb noun" format. For tasks with multiple steps, we name it as "step1+step2" format.

Evaluation Tasks and Metrics. Since our goal is to understanding the effectiveness of human-to-robot motion transfer, we focus on evaluating robot policies on the human tasks. Among all 15 tasks in human dataset, there are two tasks ("Fold Towel" and "Pour Milk Bottle") not been able to deploy to robot due to the hardware design limitation. Therefore, we focus on discussing other 13 tasks in this research. The list of all evaluated tasks can be found in Figure 5.

We use the *Success Rate (SR)* to evaluate the policy performance in accomplishing specific tasks. However, this metric alone is insufficient to reflect the effectiveness of motion transfer, as it ignores meaningful motion during task execution. To address this limitation, we define a *Motion Progress Score (Score)* to quantify the quality of policy motion for task completion. Detailed scoring rubrics for all tasks are provided in supplementary video. For clarity, we normalize the Score to a [0,1] range in the main paper. For each task, we conduct 10 rollouts and calculate the average results for both metrics.

B. Zero-shot Experiment

The goal of the zero-shot experiment is to verify the effectiveness of direct human-to-robot motion transfer. We train policies using our *MotionTrans Dataset*. Subsequently, we directly deploy policies to real robot hardware and evaluate the performance of tasks in human data. We refer to this as zero-shot setting because the policies learn motions from humans without any robot data collected for these human tasks. We seek to answer the following questions:

- (Q1.1) Can policies directly learn to accomplish tasks in human data by human-robot cotraining?
- (Q1.2) For tasks that cannot be accomplished, can the policies learn meaningful motion for task completion?
- (Q1.3) Is cotraining with robot data the key factor for achieving explicit motion transfer?
- (Q1.4) What is the difference in motion transfer effectiveness between different policy architectures?

Experiment Details. We train two end-to-end policies, Diffusion Policy (DP) and π_0-VLA (as mentioned in Section III-D). For DP, we train it for 300 epochs with a learning rate of 3×10^{-4} and 256 batch size. For π_0-VLA, we train it for 160,000 steps with a learning rate of 2.5×10^{-5} and 192 batch size. Both models are trained with the AdamW optimizer.

(Q1.1) *MotionTrans* enables policies to achieve non-trivial success rate across 9 tasks in the human dataset. The results of the zero-shot experiment are shown in Figure 5. We can see that 9 tasks achieve a non-trivial success rate. The visualization of two examples could be found in the Figure 6(a) ("Orange-Bucket" and "Unplug Charger"). Among these tasks, pick-and-place tasks account for the vast majority. This can be attributed to (1) the simplicity of pick-and-place motion and (2) the large number of such tasks in our dataset. Notably, for the cases where even if both the pick objects and place targets are not seen in robot tasks (e.g., the "Orange-Bucket" task, visualized on the left side of Figure 6(a)), this type of task-level transfer is still possible. Other accomplished tasks includes motions like pouring, unplugging, lifting, opening and closing (pressing).

(Q1.2) For unsuccessful tasks, *MotionTrans* enables policies to learn meaningful motions toward task completion. Figure 5 shows that both DP and π_0-VLA achieve positive Motion Progress Scores across all tasks, with an overall average of about 0.5. This indicates that the policies are able to complete certain sub-processes for all evaluation tasks. For instance, in the "Wipe Towel" task, both DP and π_0-VLA learn the motion of "push towel forward" to some extent (left side of Figure 6(b)). Moreover, we observe that human data enables the policy to identify spatial locations for almost all human tasks, which is represented as reaching the target manipulated objects (may only appear in human data) to some extent. An example of this is the "Press Stapler" task in Figure 6(b): although the stapler is not seen in the robot data, the policy still performs approaching behavior.

(Q1.3) Cotraining with robot data is the key factor for successful motion transfer. We find that when robot data is not included for cotraining, the success rate across all tasks is 0% for zero-shot setting. Generally, the policy trained solely on human data exhibits random motion when deployed on the robot. This demonstrates that cotraining with robot data is essential for explicit human-to-robot motion transfer, which could bridge the gap between humans and robots, allowing human motions to adapt to robot embodiment.

(Q1.4) DP and π_0-VLA each have their own advantages (manipulation precision and task adherence). As shown in Figure 5, no single model excels across all tasks. On average, the performance of the two models is nearly identical.

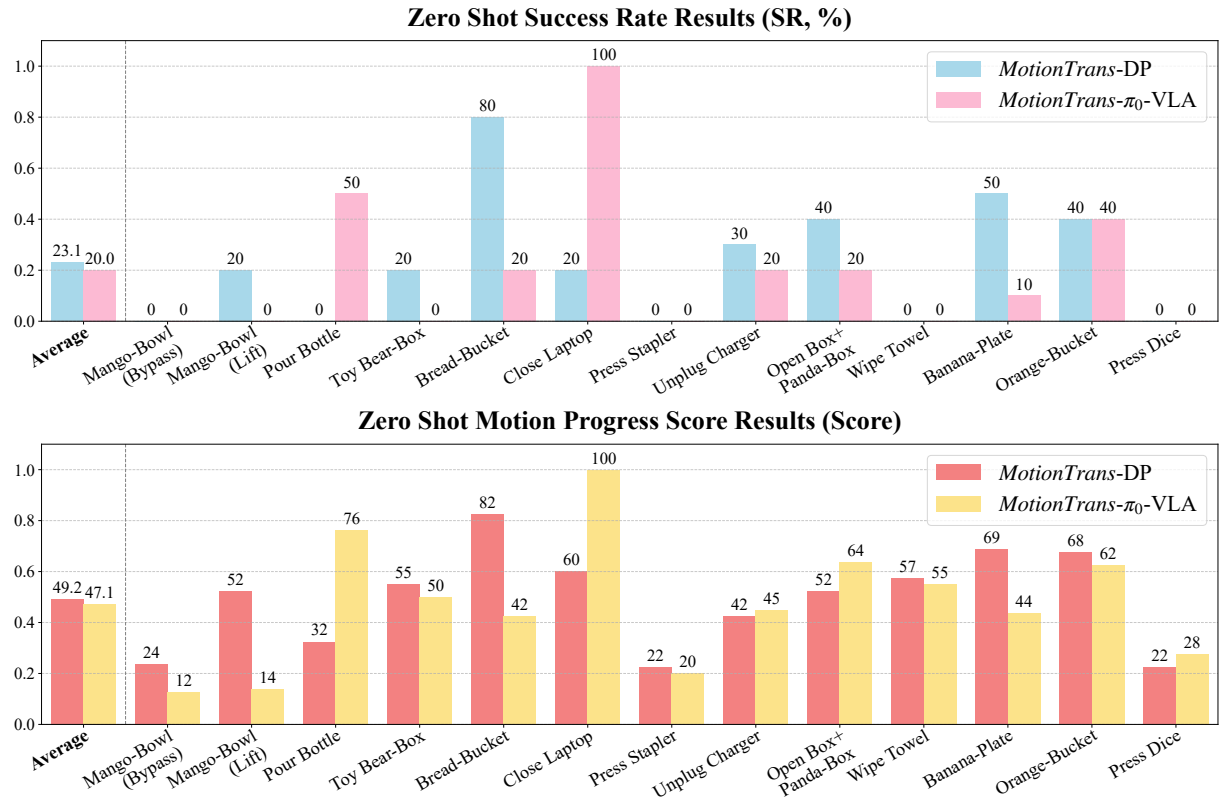

Fig. 5: Results of *MotionTrans* in the zero-shot experiment setting. The results show that both Diffusion Policy (DP) [2] and π_0-VLA [3] achieve successful human-to-robot motion transfer. Even without any robot data for these human tasks, 9 tasks attain a non-zero success rate. For the remaining tasks, *MotionTrans* still generates meaningful motion for task accomplishment, as indicated by a non-trivial Motion Progress Score.

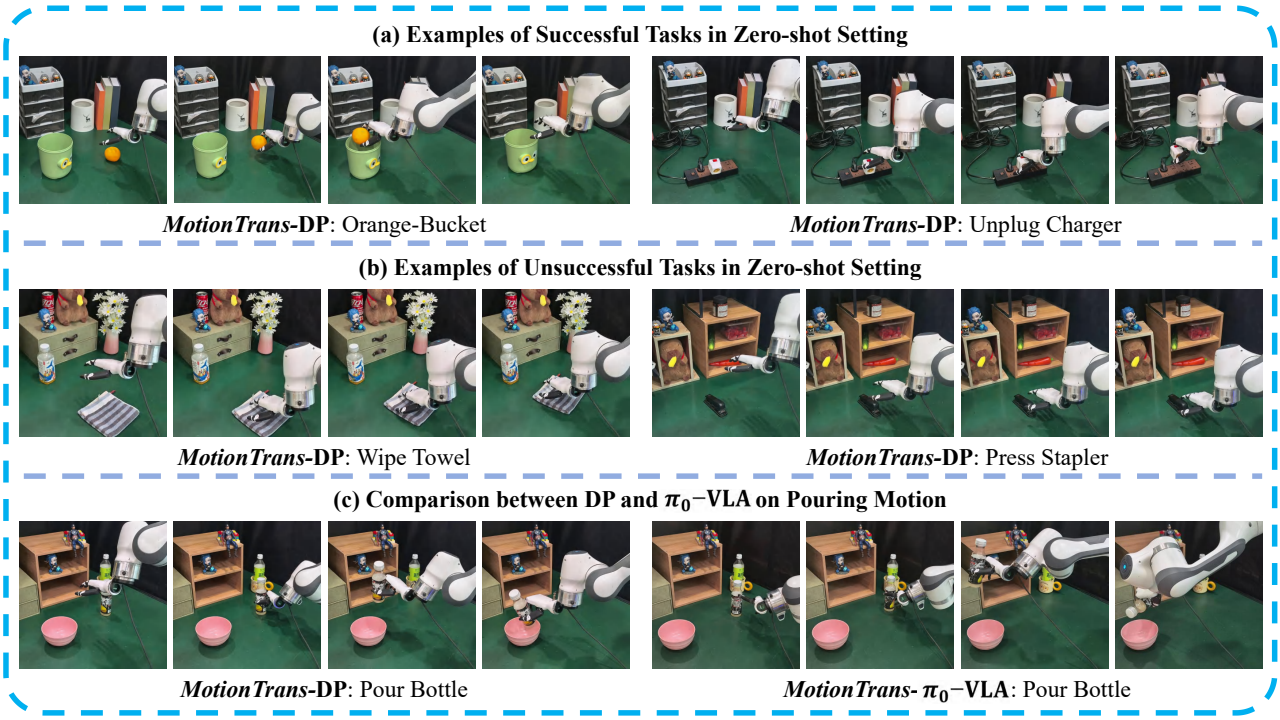

Fig. 6: The visualizations for **zero-shot** human-to-robot motion transfer from our *MotionTrans* framework. All tasks shown here do not involve any robot data collection and are learned from human data. These results demonstrate that the *MotionTrans* enables explicit human-to-robot motion transfer for task completion through human-robot cotraining.

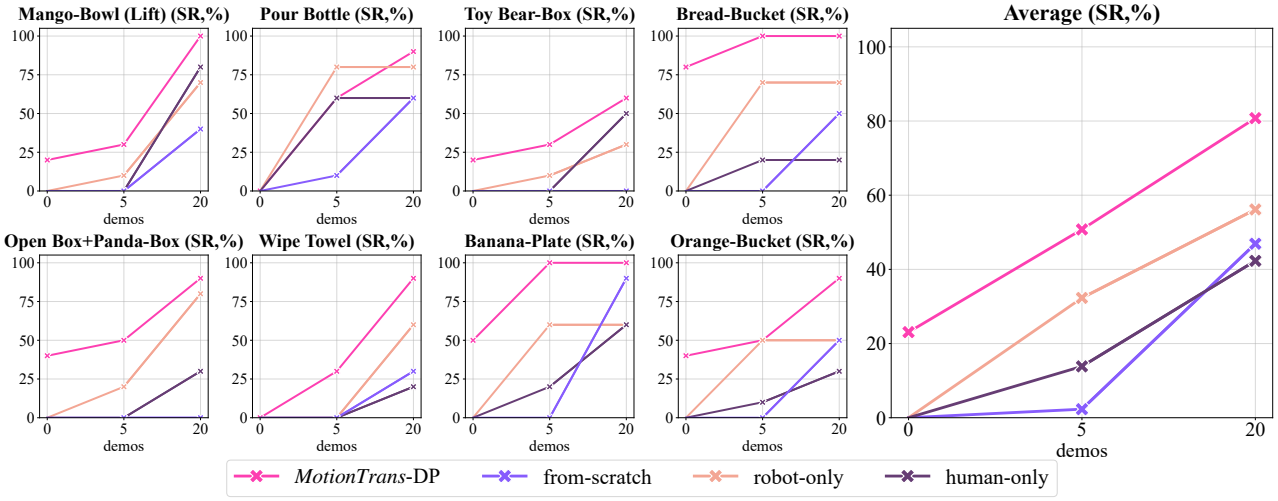

Fig. 7: Results of the success rate for few-shot finetuning experiments. For readability, only the results of 8 example tasks are presented here. From these results, we can conclude that both human and robot data during pretraining are important for improving finetuning performance.

However, we observe that different models demonstrate their strengths on different tasks. Generally, DP performs better than π_0-VLA in precise manipulation stage, such as grasping, and exhibits stronger spatial location capabilities. An example of evidence for this is that, for all pick-and-place tasks, the average grasping success rate of π_0-VLA is 20%, while DP achieves 65%. In contrast, π_0-VLA shows stronger instruction following for motion generation. For example, in the "Pour Bottle" task, we observed limited wrist rotation with DP, while π_0-VLA successfully performs the complete pouring action (Figure 6(c)). We hypothesize that the model focuses more on visual perception (DP) tends to achieve greater manipulation precision, whereas the model that emphasizes task semantics and instruction following (π_0-VLA) can adhere to task requirements more stringently.

C. *Few-shot Experiment*

In this section, we investigate whether motion transfer from human-robot cotraining can also enhance performance in a few-shot finetuning setting, where a limited number of robot demonstrations of human tasks are available for policy finetuning. We aim to answer the following questions:

- (Q2.1) Will pretraining on *MotionTrans Dataset* help improve policy finetuning performance?
- (Q2.2) What is the contribution of human data versus robot data for policy pretraining?
- (Q2.3) How does pretraining improvement vary with increasing finetuning data?

Experiment Details. Considering DP and π_0-VLA exhibit similar average performance in zero-shot experiments, we focus on DP architecture for computational resource efficiency in this part. We additionally collect 20 demonstrations for all human tasks in the evaluation scenes. Subsequently, we perform 5-shot and 20-shot **multi-task finetuning** [13] based on checkpoints previously trained on the *MotionTrans Dataset*. We finetune DP with a learning rate of 1×10^{-4} and a batch size of 256 for 200 epochs, employing the AdamW optimizer.

We compared our method with three baselines to investigate the impact of different data components: (1) **"from-scratch"**, which means training policies without pretraining; (2) **"robot-only"**, which entails pretraining solely on robot data from the *MotionTrans Dataset* before finetuning; and (3) **"human-only"**, which is pretrained exclusively on human data.

(Q2.1) Pretraining on *MotionTrans Dataset* enable significant improvement for finetuning performance. The results of the few-shot experiments are presented in Figure 7. we can see that policy pretrained on *MotionTrans Dataset* gains around 40% average success rate improvement compared to "from-scratch" baseline, proving that pretraining on human-robot data could provide useful motion prior [11] for downstream finetuning.

(Q2.2) Both robot and human data during pretraining are crucial for enhancing performance From Figure 7, we can see that policy pretrained on both human and robot data (*MotionTrans*) shows a significant advantage compared to human-only or robot-only pretraining. Besides, robot-only pretraining outperforms human-only pretraining on average. In our setting, robot pretraining uses data from the same embodiment but different tasks, whereas human pretraining uses data from the opposite case. We therefore conclude that maintaining the same embodiment in pretraining data is more important than exactly matching tasks. This is because the distribution of robot data is generally closer to the downstream robot finetuning distribution than human data, even when the tasks differ. Moreover, motions across different tasks often share similarities, so different robot tasks can still benefit downstream finetuning performance [3].

(Q2.3) Human-robot pretraining is more effective in low finetuning data region. As shown in Figure 7, the average performance of the policies improves consistently with an increase in finetuning data for all methods. However, the improvements are much larger in the 5-shot setting compared to the 20-shot setting. Moreover, when 20 finetuned demonstrations are

available, the advantage of robot-only pretraining becomes minimal, and the benefit of human-only pretraining disappears. However, in the 5-shot setting, all pretraining methods show a significant advantage over the from-scratch baseline.

V. Conclusion

In this paper, we propose *MotionTrans*, a framework that achieves motion-level learning from human data for end-to-end robot policies. The experiments show that our method achieves explicit human-to-robot motion transfer in a zero-shot setting and significantly improves finetuning performance in a few-shot setting. We hope that the new motion-centric insights that we propose could enhance the utilization of human data in robot policy learning in more effective ways. We leave the limitation discussion in *supplementary video*.

ACKNOWLEDGMENT

This work is supported by Shanghai Qi Zhi Institute Innovation Program & Spirit AI Innovation Program and the Tsinghua University Dushi Program.

References

[1] 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.

[2] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, p. 02783649241273668, 2023.

[3] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, *et al.*, "pi_0: A vision-language-action flow model for general robot control," *arXiv preprint arXiv:2410.24164*, 2024.

[4] 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.

[5] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, *et al.*, "pi_0.5: a vision-language-action model with open-world generalization," *arXiv preprint arXiv:2504.16054*, 2025.

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

[7] R. Hoque, P. Huang, D. J. Yoon, M. Sivapurapu, and J. Zhang, "Egodex: Learning dexterous manipulation from large-scale egocentric video," *arXiv preprint arXiv:2505.11709*, 2025.

[8] S. Bahl, R. Mendonca, L. Chen, U. Jain, and D. Pathak, "Affordances from human videos as a versatile representation for robotics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 778–13 790.

[9] C. Yuan, C. Wen, T. Zhang, and Y. Gao, "General flow as foundation affordance for scalable robot learning," *arXiv preprint arXiv:2401.11439*, 2024.

[10] S. Kareer, D. Patel, R. Punamiya, P. Mathur, S. Cheng, C. Wang, J. Hoffman, and D. Xu, "Egomimic: Scaling imitation learning via egocentric video," *arXiv preprint arXiv:2410.24221*, 2024.

[11] R. Yang, Q. Yu, Y. Wu, R. Yan, B. Li, A.-C. Cheng, X. Zou, Y. Fang, H. Yin, S. Liu, *et al.*, "Egovla: Learning vision-language-action models from egocentric human videos," *arXiv:2507.12440*, 2025.

[12] H. Luo, Y. Feng, W. Zhang, S. Zheng, Y. Wang, H. Yuan, J. Liu, C. Xu, Q. Jin, and Z. Lu, "Being-h0: Vision-language-action pretraining from large-scale human videos," *arXiv preprint arXiv:2507.15597*, 2025.

[13] H. Bi, L. Wu, T. Lin, H. Tan, Z. Su, H. Su, and J. Zhu, "H-rdt: Human manipulation enhanced bimanual robotic manipulation," *arXiv preprint arXiv:2507.23523*, 2025.

[14] F. Lin, Y. Hu, P. Sheng, C. Wen, J. You, and Y. Gao, "Data scaling laws in imitation learning for robotic manipulation," *arXiv preprint arXiv:2410.18647*, 2024.

[15] C. Yuan, S. Joshi, S. Zhu, H. Su, H. Zhao, and Y. Gao, "Roboengine: Plug-and-play robot data augmentation with semantic robot segmentation and background generation," *arXiv preprint arXiv:2503.18738*, 2025.

[16] S. Chen, C. Wang, K. Nguyen, L. Fei-Fei, and C. K. Liu, "Arcap: Collecting high-quality human demonstrations for robot learning with augmented reality feedback," *arXiv preprint arXiv:2410.08464*, 2024.

[17] Y. Ji, H. Tan, J. Shi, X. Hao, Y. Zhang, H. Zhang, P. Wang, M. Zhao, Y. Mu, P. An, *et al.*, "Robobrain: A unified brain model for robotic manipulation from abstract to concrete," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 1724–1734.

[18] M. Liu, M. Wang, H. Ding, Y. Xu, Y. Zhao, and Y. Wei, "Segment anything with precise interaction," in *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024, pp. 3790–3799.

[19] F. Lin, R. Nai, Y. Hu, J. You, J. Zhao, and Y. Gao, "Onetwovla: A unified vision-language-action model with adaptive reasoning," *arXiv preprint arXiv:2505.11917*, 2025.

[20] C. Yuan, G. Chen, L. Yi, and Y. Gao, "Self-supervised monocular 4d scene reconstruction for egocentric videos," *arXiv preprint arXiv:2411.09145*, 2024.

[21] 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.

[22] Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li, "Univla: Learning to act anywhere with task-centric latent actions," *arXiv preprint arXiv:2505.06111*, 2025.

[23] C.-C. Hsu, B. Wen, J. Xu, Y. Narang, X. Wang, Y. Zhu, J. Biswas, and S. Birchfield, "Spot: Se (3) pose trajectory diffusion for object-centric manipulation," *arXiv preprint arXiv:2411.00965*, 2024.

[24] H. Bharadhwaj, D. Dwibedi, A. Gupta, S. Tulsiani, C. Doersch, T. Xiao, D. Shah, F. Xia, D. Sadigh, and S. Kirmani, "Gen2act: Human video generation in novel scenarios enables generalizable robot manipulation," *arXiv preprint arXiv:2409.16283*, 2024.

[25] M. Xu, Z. Xu, Y. Xu, C. Chi, G. Wetzstein, M. Veloso, and S. Song, "Flow as the cross-domain manipulation interface," *arXiv preprint arXiv:2407.15208*, 2024.

[26] V. Liu, A. Adeniji, H. Zhan, S. Haldar, R. Bhirangi, P. Abbeel, and L. Pinto, "Egozero: Robot learning from smart glasses," *arXiv preprint arXiv:2505.20290*, 2025.

[27] M. Lepert, J. Fang, and J. Bohg, "Phantom: Training robots without robots using only human videos," *arXiv preprint arXiv:2503.00779*, 2025.

[28] M. Lepert, J. Fang, and J. Bohg., "Masquerade: Learning from in-the-wild human videos using data-editing," *arXiv preprint arXiv:2508.09976*, 2025.

[29] T. Tao, M. K. Srirama, J. J. Liu, K. Shaw, and D. Pathak, "Dexwild: Dexterous human interactions for in-the-wild robot policies," *arXiv preprint arXiv:2505.07813*, 2025.

[30] Y. Qin, W. Yang, B. Huang, K. Van Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox, "Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system," *arXiv preprint arXiv:2307.04577*, 2023.

[31] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, K. Ghasemipour, C. Finn, and A. Wahid, "Aloha unleashed: A simple recipe for robot dexterity," *arXiv preprint arXiv:2410.13126*, 2024.

[32] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[33] A. Steiner, A. S. Pinto, M. Tschannen, D. Keysers, X. Wang, Y. Bitton, A. Gritsenko, M. Minderer, A. Sherbondy, S. Long, *et al.*, "Paligemma 2: A family of versatile vlms for transfer," *arXiv preprint arXiv:2412.03555*, 2024.

[34] K. Pertsch, K. Stachowicz, B. Ichter, D. Driess, S. Nair, Q. Vuong, O. Mees, C. Finn, and S. Levine, "Fast: Efficient action tokenization for vision-language-action models," *arXiv preprint arXiv:2501.09747*, 2025.

[35] A. Wei, A. Agarwal, B. Chen, R. Bosworth, N. Pfaff, and R. Tedrake, "Empirical analysis of sim-and-real cotraining of diffusion policies for planar pushing from pixels," *arXiv preprint arXiv:2503.22634*, 2025.

[36] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024.