Learning Dexterous Manipulation from Human Video Demonstrations with 3D Point Tracks

Sungjae Park sungjae2@andrew.cmu.edu
Junkai Huang junkaih@andrew.cmu.edu
Yanbo Xu yanboxu@andrew.cmu.edu
Chaitanya Chawla cchawla@andrew.cmu.edu
Lucas Wu yuwu3@andrew.cmu.edu

Keywords: Learning from Human Video, Dexterous Manipulation

Abstract: In this paper, we explore monocular 3D point tracks as a representation for learning dexterous robot policies from human videos. 3D point tracking offers a robust and efficient intermediate representation for policy training in comparison to prior approaches using object mesh reconstruction. We validate our method in learning from a single video demonstration setup by fine-tuning the baseline on our dataset extracted from mesh-based representations and 3D point tracking approach. Our results highlight the potential of this representation to enhance policy learning from human videos.

1 Introduction

Learning directly from human data is one of the effective and accurate approaches to acquiring real-world policies. Models trained on internet-scale human data have demonstrated remarkable increases in performance across domains, including Natural Language Processing [1], Image Generation [2], [3], [4], and Speech Processing [5], [6]. Despite this progress, the field of robotics has yet to fully exploit this data exodus. Several key challenges hinder this integration: 1) **Extracting quantifiable agent actions from 2d videos** is a particularly challenging task, due to the loss of depth information, occlusion of body parts, and variability in camera perspectives. 2) **Embodiment disparity** creates a substantial gap between human and robot morphologies, making it nontrivial to map joint states directly. 3) **Environment grounding from 2D to 3D** remains an open problem, with current methods struggling to reconstruct accurate and actionable 3D representations from 2D visual data. These limitations highlight the need for innovative approaches to bridge the gap between human data and robotic manipulation.

Through this work, we tackle the problem of learning dexterous robot policy from human video demonstrations. Previous approaches extracted the meshes of the object[7] and motion of human hand[7, 8] from a demonstration video, and further used them as the supervision signal for policy training. While the pipeline is sound, the under-constrained nature of human video demonstration can make the extracted object noisy and temporally inconsistent, harming the effectiveness of policy training. Building on recent advances in point tracking, we instead propose using the monocular 3D point tracking method to track the object. Since the point tracking method is usually more robust and temporarily consistent than the mesh extraction method and also rich enough to provide a 3D position of each point in a shared global frame, we believe that it would achieve better policy-learning results.

In summary, our goal is first to reproduce the results in [7], which leverage large-scale human videos and a video reconstruction model to extract the mesh of the human hand and objects at each time step. The reconstructed hand mesh is retargeted to the robot hand, creating a pseudo-demonstration for each video. Retargeted robot hand action is used for policy pre-training along with object point

clouds. As in the paper, we show that the pre-trained policy shows superior sample efficiency for downstream task learning compared to learning from scratch. Then, we focus on the setup of learning a downstream target task from a single human video demonstration, comparing the baseline and our method.

2 Related Work

Learning from Human Video. Recently, human video data have been considered as a scalable source to acquire a generalizable robot policy. This paradigm starts with the question of what type of information to extract from human video and transfer to the robot, ranging from the reward model[9, 10], human hand articulation[11, 7], object mesh[7], high-level planner[12, 13, 14, 15], etc. Two important criteria for answering the above question are: (1) Can we reliably and robustly extract such information from general human video? (2) How rich is the information extracted that enables efficient learning of the target task? When it comes to a dexterous robot hand, a natural answer to the second question is to leverage human hand motion(i.e. wrist pose and finger articulation), along with object geometry(i.e. mesh) and motion(i.e. pose). Robot hands are anthropomorphic, having high degrees of freedom to directly mimic human hand motion. Leveraging object geometry induces rich representation, which benefits generalization across different objects, and how the object moves within the human hand specifies how the target task should be solved in detail. In this project, we focus on recent approaches that leverage this information reconstructed from monocular human videos for robot policy learning. Specifically, we use HOP [7] as our baseline method.

Reconstructing Hand-Object Interaction. Reconstructing human hands and objects from general video is a well-known topic. [16] leverage transformer to predict human hand MANO[17] parameters from a single RGB image, which can also be applied to monocular videos. On top of the human hand, [18, 19, 20] further extracts the object mesh from the video, being able to reconstruct the interaction of the human hand and the object at the mesh level. While such methods have shown interesting progress, we find that they struggle to accurately capture object geometry, and fail to show the temporally consistent motion between hand and object, given that the problem is highly under-constrained.

Point Tracking from Videos. Compared to reconstructing hand-object interactions with mesh and its motion, recent approaches considered point tracks as a reliable alternative to represent the interaction within videos. Both 2D[21] and 3D point[22, 23] tracking methods are more robust and accurate while still being able to describe the interaction between hand and object. It can also deal with interaction including deformable objects, such as cloth folding. Typically, these point tracks are combined with object segmentation masks to extract point tracks of the object being manipulated. In this project, we aim to compare 3D point tracking models with mesh reconstruction models for robot policy training from human videos.

3 Methods

3.1 Task Definition

Given a target task T, a single human video demonstration V_T for the target task, our goal is to train a dexterous robot hand policy Π_R that can solve the same task. As this is a challenging task, we additionally leverage a set of human video demonstrations $V = \{V_1, V_2, ..., V_n\}$ for policy pre-training. Note that videos in V are general human hand-object interaction videos, not necessarily solving the target task T.

3.2 Baseline Method

We follow HOP [7] as our baseline method. The overview of HOP is described in Fig.1. HOP first recovers the underlying 3D structure of hand-object interactions from in-the-wild monocular videos, in the format of MANO hand parameters for hand motion and mesh for object geometry and

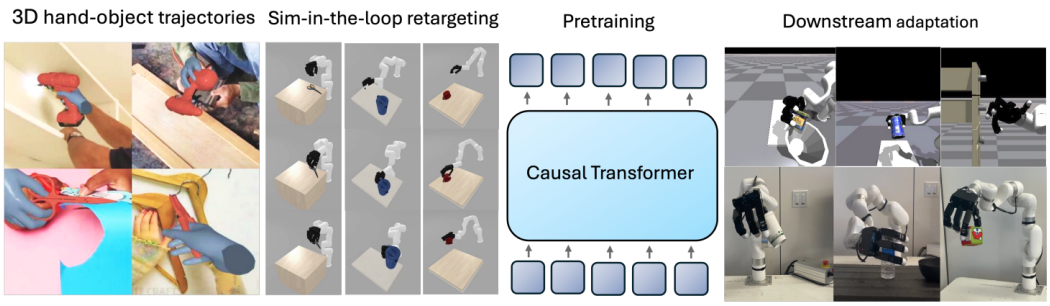


Figure 1: **Overview of Baseline**

motion. Hand motion is used to create robot hand trajectories that mimic human hand motion via inverse kinematics, and object mesh is used to generate object point clouds. The processed robot motion and object point clouds are used to pre-train the robot hand policy with behavior cloning. Finally, the pre-trained policy is fine-tuned to downstream tasks. **While the original paper assumes the target task demonstration to be provided with robotic teleoperation or applies RL for fine-tuning, we only assume a human video demonstration for the target task to be given in our task definition**. This makes the overall pipeline more convenient and scalable. As a result, the fine-tuning phase is analogous to the pre-training phase, using the given demonstration video instead of in-the-wild monocular videos.

3.2.1 Lifting Hand-Object Interaction Videos to 3D

HOP leverages HaMeR [16] and MCC-HO [18] to jointly infer hand-object geometry as point clouds. While MCC-HO includes a fine-tuning procedure with the object mesh template, HOP skips this procedure to make the approach more scalable. Instead, HOP increases temporal smoothness by anchoring object reconstructions to time-smoothed hand detections. The method also assumes that the camera from which the video is collected is static. The output of this pipeline is a sequence of 3D hand and object point clouds.

3.2.2 Mapping 3D Human-Object Interactions to Robot-Object Interactions

HOP maps the hand trajectories to robot trajectories by formulating a non-linear optimization problem with inverse kinematics. At each step k, they find the robot action $\mathbf{a}[k]$ by optimizing the following cost function:

$$\min_{\boldsymbol{a}[k]} \frac{1}{2} \left\| \boldsymbol{x}_h[k] - f(\boldsymbol{a}[k]) \right\|^2 + \lambda \|\boldsymbol{a}[k] - \boldsymbol{\phi}[k-1]\|^2 \quad \text{s.t.} \quad \boldsymbol{a}[k] \in \mathbb{A}, \tag{1}$$

where f is the robot's forward kinematics, and $\mathbf{x}_h[k]$ are the 3D coordinates of a set of keypoints on the human hand. The first term of Eq. 1 represents the difference between the keypoints of the robot and the human hand as a function of the desired joints of the robot $\mathbf{a}[k]$. The second term is proportional to the energy required to execute the action $\mathbf{a}[k]$, which is to minimize to favor smoothness. For keypoints, we use fingertips of the human hand and robot hand.

3.2.3 Robot Policy Pretraining & Finetuning

After obtaining the hand-object trajectory dataset $\mathcal{T}$, HOP incorporates the prior knowledge in the dataset $\mathcal{T}$ (*i.e.*, where and how to grasp; some intuitive physics; wrist-hand coordination, *etc.*) into a policy π_b that can be fine-tuned to downstream tasks. [7] instantiates π_b as a transformer and train it to capture the conditional distribution $\Pi(\boldsymbol{a}[t-L:t] \mid \boldsymbol{o}[t-L:t])$ by optimizing the following loss, where o corresponds to object point clouds or RGBD images.

$$\mathcal{L}(\tau;\theta) = \mathbb{E}_{t\sim[1\ldots T]} \left[\|\boldsymbol{a}[t-L:t] - \pi_b(\boldsymbol{o}[t-L:t])\|_1 \right]. \tag{2}$$

The pretrained policy π_b does exhibit primitive manipulations skills like reaching an object wiht a reasonable grasp pose, while occasionally grasping successfully. The policy is also fine-tuned

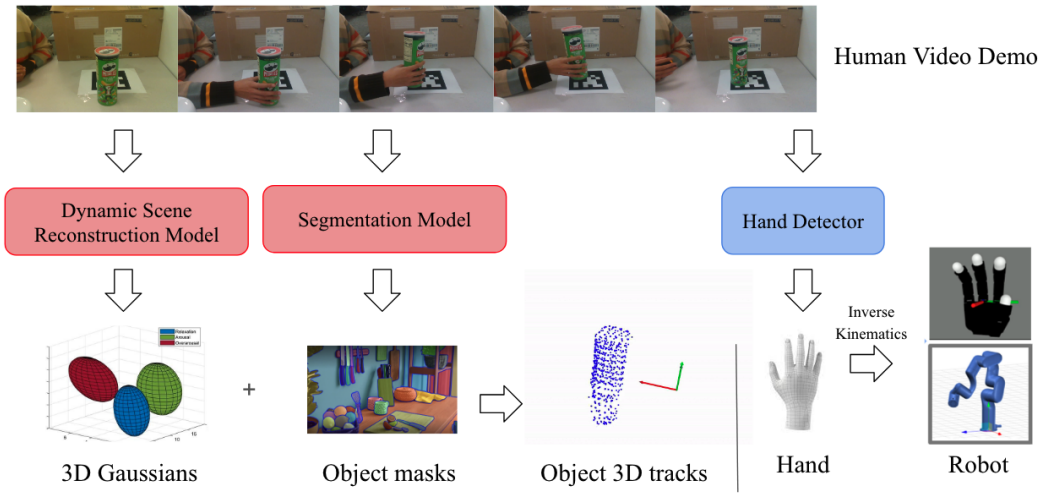


Figure 2: **Overview of Proposed Pipeline**

for different tasks like grasp & lift, lift & throw, and open cabinet. According to the authors, the pretrained model can be finetuned much easier than initialized from scratch. The experimental results reproduced by us (shown in Sec. 4.2) corroborate the conclusion.

3.3 Proposed Method

We find that the object mesh and the point clouds inferred from MCC-HO are inaccurate and temporally inconsistent, even when given with interaction between hand and rigid objects. As a result, we instead leverage the state-of-the-art 4D reconstruction method, Shape-of-Motion[23], to extract 3D point tracks and compute object point clouds from them.

Specifically, Shape-of-Motion fits a set of 3D Gaussians to reconstruct the given video up to RGB. The fitted 3D Gaussians can be used to compute the 3D point track of arbitrary query points. Compared to other 3D point tracking methods, Shape of Motion shows more robust and accurate 3D point tracking results. The 3D Gaussians can be further filtered with object masks to get 3D Gaussians corresponding to the object being manipulated, and its 3D point tracks. We use Track-Anything[24] to obtain object masks. The overview of the proposed method is shown in Fig.2. For hand detection, we use the same model HaMeR [16] as the baseline.

4 Experiments

4.1 Experiment Setup

We first aim to reproduce the HOP's results(Sec.4.2). Specifically, **we reproduce downstream task learning results in HOP, by leveraging RL in simulation as proposed in the original paper.** For downstream tasks, we choose the following: (1) Grasp and Lifting and (2) Opening the cabinet. We use the provided pre-trained policy checkpoint.

Next, we focus on the aforementioned task definition of **downstream task learning from a single human video demonstration**(Sec.4.3), where we compare the baseline and our pipeline. While one can also pre-train the policy on large-scale videos with our pipeline, we used the checkpoint provided by HOP due to the lack of computation. We choose pick and placing the bottle as the target task.

4.2 Reproducing Baseline Results

Here we reproduce the results presented in the HOP paper. They align with what has been reported in the paper. Note that the RL agent fails to learn the downstream task without pre-training, while the pre-trained policy successfully learns the task. This shows that policy pre-training from human videos has learned meaningful prior, such as reaching the object.

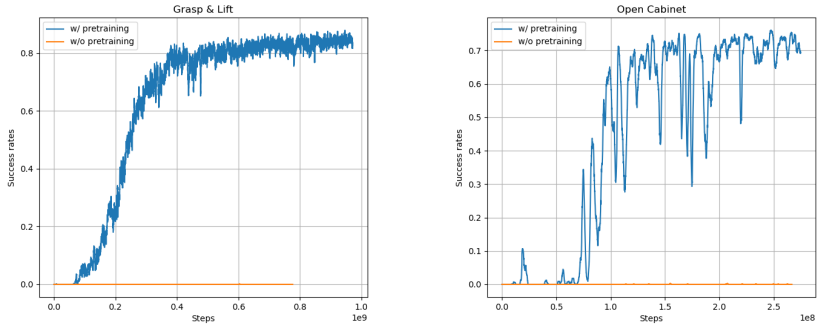
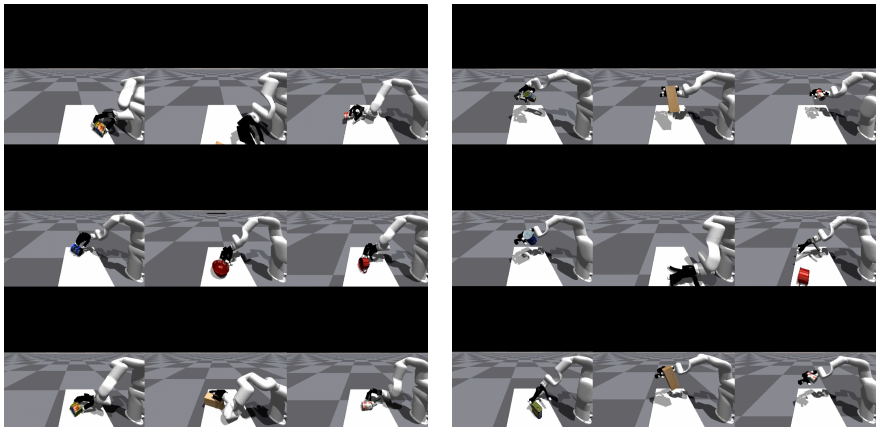


Figure 3: **Comparison of video-pretrained actor with non-pretrained baseline**. Video pretraining improves sample-efficiency of online RL across multiple tasks, particularly when the downstream task and the behaviors in the data are less aligned.



Pretrain | Pretrain + Finetune

Figure 4: **Qualitative results of pretrained actor and pretrained & finetuned actor on the Grasp & Lift task**. One can observe that the pretrained actor can already approach the target object with a good grasping pose, while finetuned model can successfully grasp and lift the target object with a high success rate.

4.3 Target Task Finetuning from Single Human Video Demonstration

We collect a single human video demonstration of picking and placing the bottle, as shown in Fig.5. By extracting the object point tracking and the human hand mesh, we utilize inverse kinematics to retarget the human hand to the robot. This process creates a pseudo-ground truth for the robot action. By fine-tuning the policy trained on a large amount of online data with the new "demonstration", we can imitate the picking and placing action with the robot, as shown in Fig.5. The qualitative comparison to the baseline for object reconstruction is in Fig.6. As shown in the figure, using 3D point tracks shows qualitatively more accurate and consistent results compared to the baseline for reconstructing the bottle. For the fine-tuned policy, we found that both baseline and our method

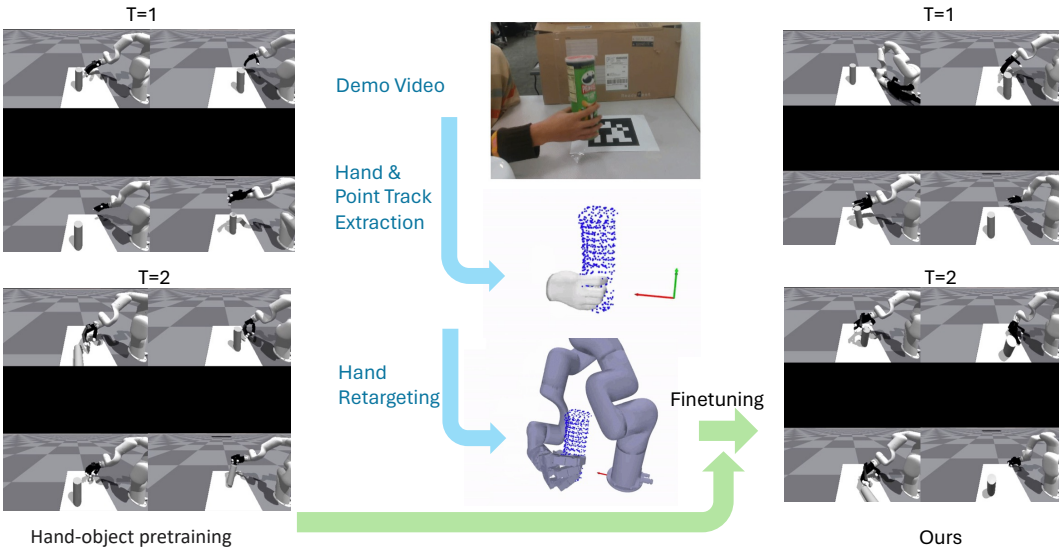


Figure 5: **Policy Finetuning using Human Video Demonstration.** The left column shows the action of pretrained policy on 4 random initialized bottle position. The right column displays the action after fintuning using our demonstration.

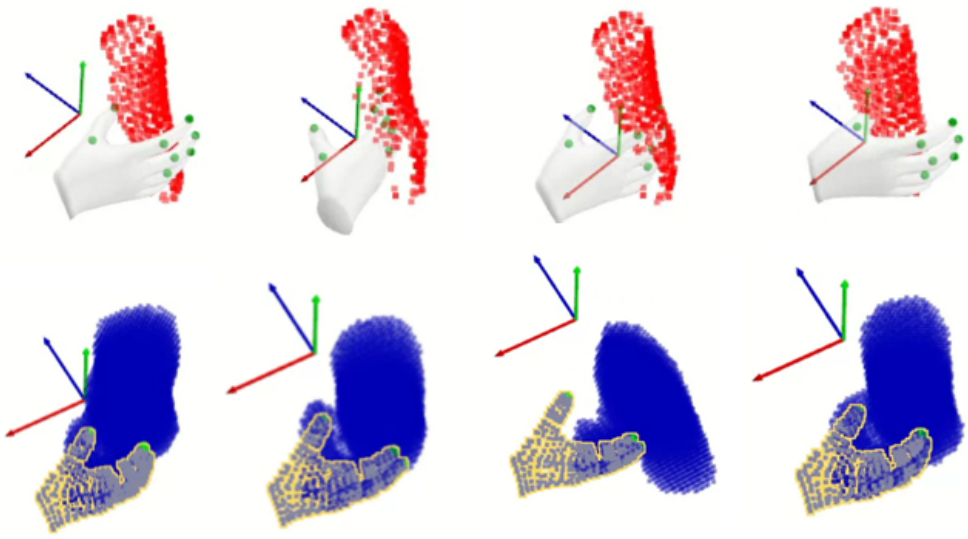


Figure 6: **Reconstructed Hand Mesh and Object Point Clouds.** Top row is from ours and the bottom row is from baseline.

result in zero success rate, not being able to grasp and pick up the bottle. There are multiple reasons of failure. First, while the reconstructed object is qualitative better with our pipeline, it is still far from perfect, showing penetration with extracted human hand in the middle of interaction. Hence, we observed the trained robot policy tripping the bottle. Additionally, we assumed matching fingertip positions between human hand and robot hand would result in same interaction, while this may not hold in general due to embodiment gap. Lastly, as we only used single video demonstration, the lack of target task data would have caused failure.

5 Conclusion

In this project, we proposed using 3d point tracks as object state representations from human video demonstrations as it is more general, flexible, and doesn't require prior knowledge of the object that we are interacting with. We recorded our own human demonstration video, successfully extracted 3D point tracks on the object, reconstructed human hand poses and mapped them to robot trajectories. Although fine-tuning the robot policy on our data wasn't successful because of the imperfection of 3D point tracks and the lack of diversified training data that we can obtain within limited amount of time, we believe running the current pipeline for large-scale policy pre-training from human videos would be a promising future direction to pursue, as the policy would still learn meaningful behaviors during pre-training. Additionally, leveraging reactive feedback control would help in learning from noisy demonstrations, which are common due to imperfect reconstruction modules.

References

[1] T. B. Brown. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[2] A. Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[3] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[4] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr, 2021.

[5] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR, 2023.

[6] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

[7] H. G. Singh, A. Loquercio, C. Sferrazza, J. Wu, H. Qi, P. Abbeel, and J. Malik. Hand-object interaction pretraining from videos. *arXiv preprint arXiv:2409.08273*, 2024.

[8] Y. Qin, Y.-H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang. Dexmv: Imitation learning for dexterous manipulation from human videos. In *European Conference on Computer Vision*, pages 570–587. Springer, 2022.

[9] K. Zakka, A. Zeng, P. Florence, J. Tompson, J. Bohg, and D. Dwibedi. Xirl: Cross-embodiment inverse reinforcement learning. In *Conference on Robot Learning*, pages 537–546. PMLR, 2022.

[10] S. Kumar, J. Zamora, N. Hansen, R. Jangir, and X. Wang. Graph inverse reinforcement learning from diverse videos. In *Conference on Robot Learning*, pages 55–66. PMLR, 2023.

[11] K. Shaw, S. Bahl, and D. Pathak. Videodex: Learning dexterity from internet videos. In *Conference on Robot Learning*, pages 654–665. PMLR, 2023.

[12] H. Bharadhwaj, A. Gupta, and S. Tulsiani. Visual affordance prediction for guiding robot exploration. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3029–3036. IEEE, 2023.

[13] H. Bharadhwaj, A. Gupta, V. Kumar, and S. Tulsiani. Towards generalizable zero-shot manipulation via translating human interaction plans. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6904–6911. IEEE, 2024.

[14] H. Bharadhwaj, R. Mottaghi, A. Gupta, and S. Tulsiani. Track2act: Predicting point tracks from internet videos enables diverse zero-shot robot manipulation. *arXiv preprint arXiv:2405.01527*, 2024.

[15] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. *arXiv preprint arXiv:2302.12422*, 2023.

[16] 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*, pages 9826–9836, 2024.

[17] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *arXiv preprint arXiv:2201.02610*, 2022.

[18] J. Wu, G. Pavlakos, G. Gkioxari, and J. Malik. Reconstructing hand-held objects in 3d. *arXiv preprint arXiv:2404.06507*, 2024.

[19] A. H.-O. Prior. G-hop: Generative hand-object prior for interaction reconstruction and grasp synthesis supplementary material.

[20] Z. Fan, M. Parelli, M. E. Kadoglou, X. Chen, M. Kocabas, M. J. Black, and O. Hilliges. Hold: Category-agnostic 3d reconstruction of interacting hands and objects from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 494–504, 2024.

[21] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira, and A. Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10061–10072, 2023.

[22] Y. Xiao, Q. Wang, S. Zhang, N. Xue, S. Peng, Y. Shen, and X. Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20406–20417, 2024.

[23] Q. Wang, V. Ye, H. Gao, J. Austin, Z. Li, and A. Kanazawa. Shape of motion: 4d reconstruction from a single video. *arXiv preprint arXiv:2407.13764*, 2024.

[24] J. Yang, M. Gao, Z. Li, S. Gao, F. Wang, and F. Zheng. Track anything: Segment anything meets videos, 2023.