Open-TeleVision: Teleoperation with Immersive Active Visual Feedback

Anonymous Author(s)
Affiliation
Address
email

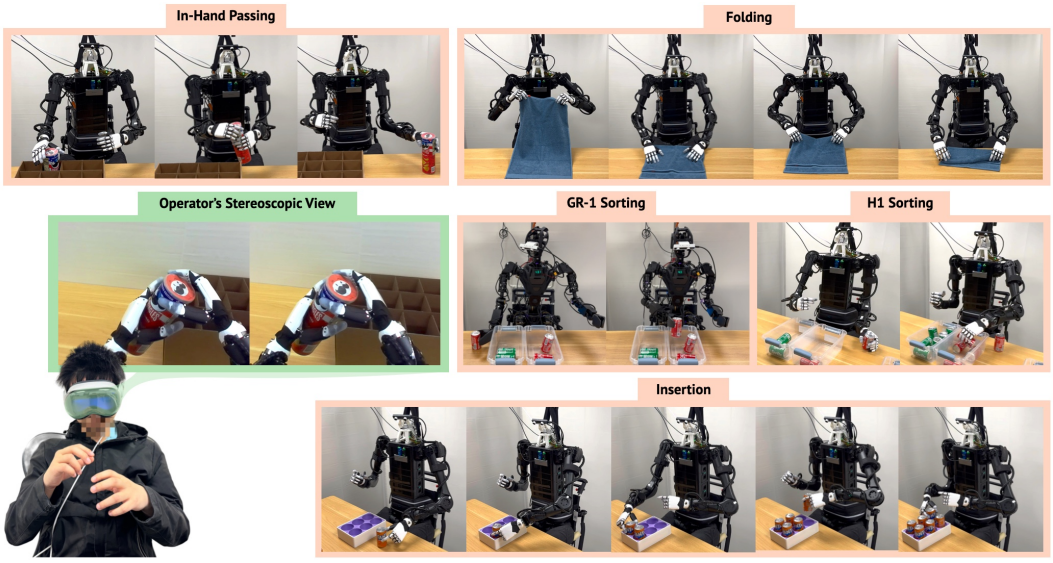


Figure 1: We present a framework for immersive robot teleoperation. Green: the robot follows the operator's head, hands and arms' movements to observe and interact with its surroundings. The operator observes an immersive stereoscopic video stream from the robot's camera. Red: 2 different humanoid robots (GR-1 and H1) autonomously execute tasks via imitation learning from data collected with our system.

Abstract: Teleoperation serves as a powerful method for collecting on-robot data essential for robot learning from demonstrations. The intuitiveness and ease of use of the teleoperation system are crucial for ensuring high-quality, diverse, and scalable data. To achieve this, we propose an immersive teleoperation system **Open-TeleVision** that allows operators to actively perceive the robot's surroundings in a stereoscopic manner. Additionally, the system mirrors the operator's arm and hand movements on the robot, creating an immersive experience as if the operator's mind is transmitted to a robot embodiment. We validate the effectiveness of our system by collecting data and training imitation learning policies on four long-horizon, precise tasks (*can sorting*, *can insertion*, *folding*, and *unloading*) for 2 different humanoid robots and deploy them in the real world. The entire system will be open-sourced. Website: https://open-television.github.io/

Keywords: Teleoperation, VR/AR, Imitation Learning

1 Introduction

Learning-based robotic manipulation has advanced to a new level in the past few years, thanks to large-scale real-robot data [1, 2]. Teleoperation has been playing an important role in data collection, where it not only offers accurate and precise manipulation demonstrations, but also provides natural

Submitted to the 8th Conference on Robot Learning (CoRL 2024). Do not distribute.

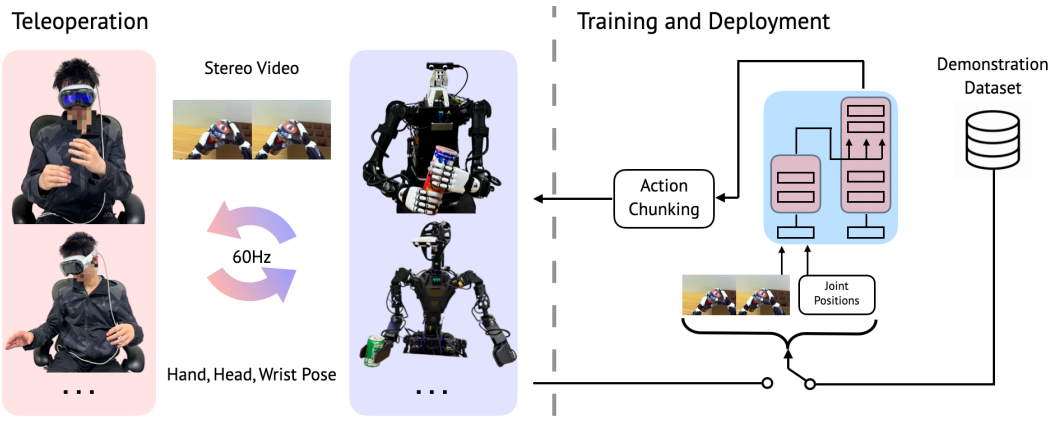


Figure 2: **Left**: our teleoperation system. VR devices stream the hand, head, and wrist poses to the server. The server retargets the human poses to the robot and sends joint position targets to the robot. **Right**: we train an imitation policy for each task with action chunking transformer. The transformer encoder captures the relationship of image and proprioception tokens and the transformer decoder outputs action sequences of a certain chunk size.

and smooth trajectories which allows the learned policies to generalize better. Various teleoperation approaches have been studied including using VR devices [3, 2], RGB cameras [4, 5, 6], wearable gloves [7, 8, 9], and customized hardwares [10, 11].

There are two major components of the teleoperation system: actuation and perception. For actuation, using joint copy to puppeteer the robot provides high control bandwidth and precision [10, 12, 13]. However, this requires the operators and the robot to be physically in the same location, not allowing for remote control. Each robot hardware will need to be coupled with one specific teleoperation hardware, reducing the generalizability of the framework. Importantly, these systems are not able to operate multi-finger dexterous hands yet. For perception, the most straightforward way is to directly observe the robot task space with the operator's own eyes in a third-person view [5, 4, 3] or a first-person view [14, 15]. This inevitably will cause occlusion on the operator's sight during teleoperation (e.g., occluded by robot arms or torso), and the operator cannot ensure the collected demonstration has captured the visual observation needed for policy learning. Displaying a third-person static camera view or using passthrough in the VR headset [3, 2, 16] encounter similar challenges.

In this paper, we propose to revisit teleoperation systems with VR devices. We introduce OpenTeleVision, providing an open and general framework to operate with high precision applicable to different VR devices on different robots and manipulators. For actuation, we operate 2 humanoid robots including Unitree H1 [17] humanoid robot with multi-finger hands and Fourier GR1 [18] humanoid robot with grippers, on bimanual manipulation tasks as shown in Fig. 1. With the captured human operators' hand pose, we can perform re-targeting to control multi-finger robot hands or parallel-jaw grippers. We rely on inverse kinematics to convert the operator's hand root position to the robot arm end-effector position.

Our major innovation comes from perception, incorporating the VR systems with active visual feedback. We use a single active stereo RGB camera on the robot head equipped with 2 or 3 DoFs actuation, mimicking human head movement to observe a large workspace. During teleoperation, the camera moves with the operator's head, streaming real-time, ego-centric 3D observations to the VR device. Specifically, with this **first-person active sensing**, we align the robot and human perception, helping the robot to react to the visual observations like humans do, significantly improving the policy: the recorded demonstrations come with both the active camera inputs and the action of how the camera moves. Our policy will learn to move the camera actively as learning manipulation. Learning active vision naturally provides an attention mechanism for performing precise manipulation. **Stereoscopic video is streamed** from robot view to human eyes. The operator can have a better spatial understanding which is shown to be crucial for completing tasks in our experiments. We also show training with stereo image frames also increases the performance of the policy.

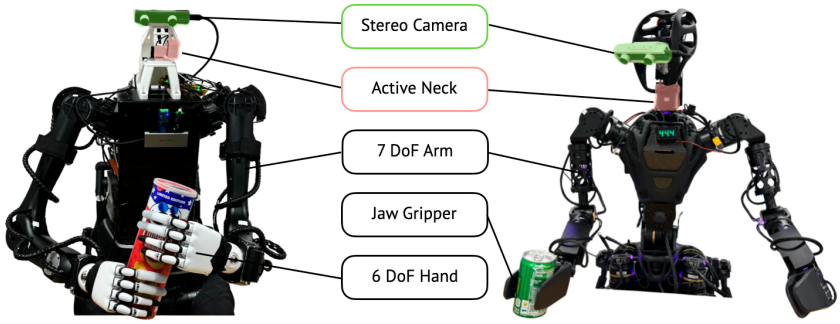


Figure 3: Our system design on two different humanoid robots. Left: Unitree H1 [17] with 6 DoF Inspire [19] hands. The active neck is composed of one yaw and one pitch motor. Right: Fourier GR-1 [18] with jaw gripper. The active neck is from the manufacturer with yaw roll and pitch motors.

We show through extensive real-world imitation learning that the data collected with our system is of high-quality and is readily applicable to precise, long-horizon bimanual tasks. We also conduct ablation and user studies to show the importance of stereo perception.

2 TeleVision System

Our system overview is shown in Fig. 2. We develop a web server based on Vuer [20]. The VR devices stream the operator's hand, head and wrist poses in $\mathrm{SE}(3)$ to the server, which handles the human-to-robot motion retargeting. In turn, the robot streams stereo video at a resolution of 480x640 for each eye. The entire loop happens at 60 Hz. While our system is agnostic to VR device model, we choose Apple VisionPro as our VR device platform in this paper.

Hardwares: For robots we choose 2 humanoid robots as shown in Fig. 3: Unitree H1 [17] and Fourier GR-1 [18]. We only consider their active sensing neck, two 7DoF arms and end-effectors, while other DoFs are not used. H1 has 6 DoFs for each hand from [19], while GR-1 has a 1 DoF jaw gripper. For active sensing, we design a gimbal with two revolute DoFs (yaw and pitch) mounted on top of the torso of H1. The gimbal is assembled from 3D printed parts and powered by DYNAMIXEL XL330-M288-T motors [21]. For GR-1, we use the 3 DoF neck (yaw, roll, and pitch) that comes with the manufacturer. A ZED Mini [22] stereo camera is used with both robots to provide stereo RGB streaming. We mostly feature humanoid robots in our setup because the teleoperation issue of occlusion and lack of intuitiveness of current system is the most prominent. While our system is specifically tailored for humanoid robots to maximize the capability for immersive teleoperation experiences, it is versatile enough to be applied to any setup featuring two arms and one camera.

Arm Control: The human wrist poses are first converted into the robot's coordinate frame. Specifically, the relative positions between the robot end-effectors and the robot head are expected to match those between the human wrists and head. The orientations of the robot wrists are aligned with the absolute orientations of the human wrists, as estimated during the initialization of the Apple VisionPro hand tracking backend. This differentiated treatment of end-effectors' positions and orientations ensures the stability of the robot end-effectors when the robot's head moves along with human's head. we employ the Closed-loop Inverse Kinematics (CLIK) algorithm based on Pinocchio [23, 24, 25] to compute the joint angles of the robot's arm. The input end-effector poses are smoothed using an $\mathrm{SE}(3)$ group filter, implemented with Pinocchio's $\mathrm{SE}(3)$ interpolation, enhancing the stability of the IK algorithm. To further mitigate the risk of IK failures, our implementation incorporates a joint angle offset when the arm's manipulability approaches its limits. This correction procedure has minimal impact on end-effector tracking performance, as the offset is projected onto the nullspace of the robot arm's Jacobian matrix, thereby preserving tracking accuracy while addressing the constraints.

Hand Control: The human hand keypoints are translated into robot joint angle commands through dex-retargeting, a highly versatile and fast-computing motion retargeting library [4]. Our approach utilizes vector optimizers on both dexterous hand and gripper morphologies. The vector optimizers

formulate the retargeting problem as an optimization problem [4, 26] while the optimization is defined based on user-selected vectors:

$$\min_{q_t} \sum_{i=0}^{N} ||\alpha v_t^i - f_i(q_t)||^2 + \beta ||q_t - q_{t-1}||^2. \tag{1}$$

In the above formulation, q_t denotes the robot joint angles at time t, and v_t^i is the i-th keypoint vector on the human hand. The function $f_i(q_t)$ computes i-th keypoint vector on the robot hand using forward kinematics from the joint angles q_t. The parameter α is a scaling factor that accounts for the hand size difference between the human hand and the robot hand (we set it as 1.1 for Inspire hands). The parameter β weights the penalty term that ensures temporal consistency between consecutive steps. The optimization is conducted in real-time using Sequential Least-Squares Quadratic Programming (SLSQP) algorithm [27] as implemented in NLopt library [28]. The computation of forward kinematics and its derivatives are conducted in Pinocchio [24].

For dexterous hands, we employ seven vectors to synchronize the human and robot hands: five vectors represent the relative positions between the wrist and each fingertip keypoint; two additional vectors, spanning from the thumb fingertip to the primary fingertips (index and middle), are incorporated to enhance the motion accuracy during fine-grained tasks. For grippers, optimization is achieved using a single vector, defined between the human thumb and index fingertips. This vector is aligned with the relative position between the gripper's upper and lower ends, enabling intuitive control over the grippers opening and closing motions by simply pinching the operator's index and thumb fingers.

3 Experiments

In this section, we aim to answer the following questions:

- How the key design choices of our system affect the performance of imitation learning results?
- How effective is our teleoperation system in collecting data?

We choose ACT [10] as our imitation learning algorithm with two key modifications. First, we replace the ResNet with a more powerful visual backbone DinoV2 [29, 30], a pretrained vision transformer (ViT) by self-supervised learning. Second, we use two stereo images instead of four images from individually arranged RGB cameras as the input to the transformer encoder. The DinoV2 backbone produces 16×22 tokens for each image. The state token is projected from the current joint position of the robots. We use absolute joint positions as the action space. For H1 the action dimension is 28 (7 for each arm, 6 for each hand, and 2 for active neck). For GR-1 the action dimension is 19 (7 for each arm, 1 for each gripper, 3 for active neck). The proprioception token is projected from the corresponding joint position readings.

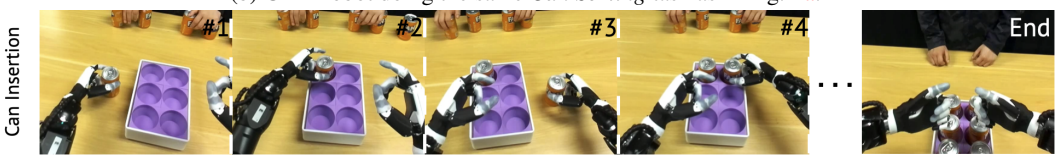
We choose four tasks with an emphasis on precision, generalization and long horizon to show the effectiveness of our proposed teleoperation system in Fig. 4. These tasks require actively looking to the left or right of the workspace using a single active camera, otherwise a multi-camera setup on the table. They also involve significant object location randomization and different manipulation strategies. We further show the data collection intuitiveness and speed by conducting user studies on all four tasks and comparing them with humans doing the same tasks. The tasks include:

Can Sorting: This task as shown in Fig. 4a involves sorting randomly placed Coke cans (red) and Sprite cans (green) on a table. The cans are placed on the table one-by-one but with random positions and types (Coke or Sprite). The goal is to pick up each can on the table and toss it into the designated case: left for Coke and right for Sprite. Solving this task requires the robot to adaptively generalize upon the position and orientation of each can for accurate grasping. It also demands that the policy can adjust its planned motions based on the color of the can it is currently holding. Each episode consists of sorting 10 cans (5 Sprite 5 Coke randomly) consecutively.

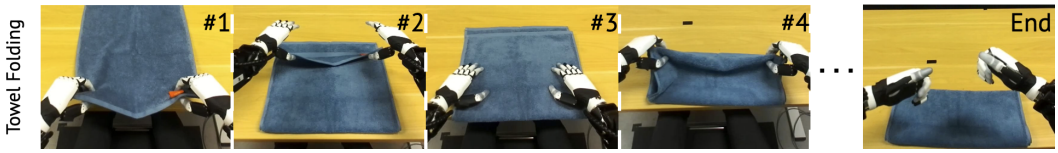
Can Insertion: This task shown in Fig. 4c involves picking up soft drink cans from the table and carefully inserting them into slots within a container in a predefined sequence. While both involve

(a) The robot picks up a randomly placed can (*#1*), places the red can in the left case (*#2*), picks up the next can (*#3*), places the green can in the right case (*#4*). When the sorting box at the edge of the table is empty (as shown in *#1, #3*) and there is no more can on the table, the robot poses the ending gesture (*End*).

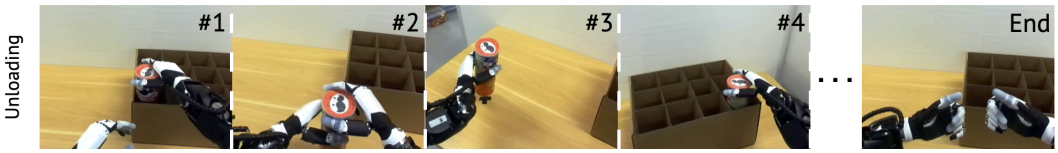
(b) GR-1 robot doing the same *Can Sorting* task as in Fig. 4a.



(c) The robot picks up the can (*#1*), inserts the can into the slot (*#2*), uses the other hand to repeat picking and inserting(*#3, #4*). When all six slots have been filled, the robot poses the ending gesture (*End*).



(d) The robot picks up the two corners of the towel (*#1*), folds the towel (*#2*), gently moves the towel to the edge of the table (*#3*), folds the towel twice (*#4*). When the towel has been folded into a satisfactory configuration, the robot poses the ending gesture (*End*).



(e) The robot extracts the tube from a random slot (*#1, #4*), passes from right hand to left (*#2*), places the tube in the designated position (*#3*). When there is not a tube anymore, the robot poses the ending gesture (*End*).

Figure 4: Demonstration collection process of H1 for four tasks. Each roll represents an individual task execution. At the end of each task, the operator postures to the same ending gesture to signify the successful completion of one demonstration. Then our system will stop recording data. We do not crop the ending gesture from our dataset. The learned policy can successfully react to ending conditions.

manipulation of drink cans, this task demands more precise and fine-grained actions than the previous one as a successful insertion necessitates high accuracy. In addition, a different grasping strategy is adopted in this task. In the previous can-sorting task, the robot only needs to toss the can into a designated area, hence we form a grasp that involves the palm and all five fingers, which is a tolerant but imprecise grasping strategy. In this one, to insert the can into a slot that is only slightly larger than the can (the diameter of the soda can is roughly 5.6 cm, and the diameter of the slot is roughly 7.6 cm), we employed a more pinching-like strategy that utilizes only the thumb and index fingers, enabling more granular adjustments in the placement of the cans. The two distinct grasping strategies demonstrate that our system is able to accomplish tasks with complex hand gesture requirements. Each episode of this task include picking and placing all six cans into the right slots.

Folding: This task shown in Fig. 4d involves folding the towel twice. The distinction of task is that it manifests the system's capability to manipulate soft and compliant materials like a towel. The action sequence of this task unrolls as follows: pinching the two corners of the towel; lifting and folding;

Baselines	H1 Can Sorting		GR-1 Can Sorting		Can Insertion	
	Pick	Place	Pick	Place	Pick	Insert
Ours	0.92	0.88	0.87	0.60	0.90	0.87
ResNet	0.74	0.58	0.83	0.50	0.53	0.70
Mono	0.46	0.52	0.73	0.63	0.47	0.63

Baselines	Folding				Unloading		
	Lift	Fold	Move	Fold	Extract	Pass	Place
Ours	1.00	1.00	1.00	1.00	1.00	1.00	1.00
ResNet	1.00	1.00	1.00	1.00	0.85	1.00	0.95
Mono	1.00	1.00	0.60	1.00	0.70	0.95	1.00

Table 1: Success rate for each sub-task. We record 5 real-world episodes for each task. Each episode contains a complete task cycle defined in Sec. 3. On GR-1, each *Can Sorting* roll-out contains 6 pickings and 6 placings, accumulating 30 trials for each sub-task.

gently moving the towel to the edge of the table to prepare for a second fold; pinching, lifting, and folding again. Each episode of this task consists of one complete folding of the towel.

Unloading: This task shown in Fig. 4e is a composite operation that involves tube extraction followed by in-hand passing. In this task, a chip tube is randomly placed into one the four slots within a sorting box. The goal is to identify the slot containing the tube, extract the tube using the right hand, pass it to the left hand and place the tube in a predefined location. To successfully execute this task, the robot needs both visual reasoning to discern the tube's location and accurate action coordination for extraction and in-hand passing. Each episode of this task consists of picking up 4 tubes from 4 random slots, passing them to another hand, and finally putting them on the table.

3.1 Imitation Learning Results

We ablate our key design choices of the system and show from real-world experiments their effectiveness. The two baselines are *ResNet*, which uses the original ACT visual backbone ResNet18 [31], and *Mono*, which only takes the visual tokens form the left image instead of both. All models are trained using AdamW optimizer [32, 33] with a learning rate of $5e-5$, a batch size of 45 and for $25k$ iterations on a single RTX 4090 GPU. We conducted *Can Sorting* task on both H1 and GR-1 robots. All other tasks are conducted with H1 robot only.

From our experiment results shown in Tab. 1, we have noticed that the original ResNet backbone fails to adequately perform all four tasks. In the original ACT implementation, four cameras are used (two fixed, two wrist mounted) to alleviate the deficiency of explicit spatial information from using RGB images. However, our setup involves only two stereo RGB cameras, potentially making spatial information retrieval more difficult for ResNet backbone.

Can Sorting: We evaluate the success rate of picking up the can and accuracy of placing it in the designated case separately. According to the results on H1 in Tab. 1, our model has the highest success rate in both evaluation metrics. *ResNet* is distinctly inferior in both picking and sorting, likely due to its backbone's limitations. Without implicit depth information from stereo inputs, *Mono* fails to properly pick up the can (23/50 success rate). Its low sorting accuracy (26/50 success rate) is highly correlated to its poor performance in the previous stage, as an experimenter must frequently help it to grasp the can, an action that interferes with visual inference.

The results of *Can Sorting* conducted on GR-1 are also reported in Tab. 1. In the picking sub-task, our model consistently outperforms the other two baselines. However, in the placing sub-task, none of the models have reached satisfactory accuracy. This phenomenon can be attributed to the different morphologies between the dexterous hand and the gripper: when a robot hand grasps the can, the camera is able to see the color of the can and make its actions based on what it sees; in contrast, when a gripper grasps the can, the camera can barely make out the color due to significant

occlusion(Fig. 4b), complicating visual inference. The other limitation is that ACT's ability to make long-horizon inference is dependent on its chunk size. In our setup, using a chunk size of 60 with an inference frequency of 60 Hz effectively provides the robot with one second of memory: when the robot is supposed to drop the can without direct visual confirmation, it has likely forgotten the color of the can which was visible at the time of pickup.

Can Insertion: We evaluate the success rate of picking up and insertion separately. The results (Tab. 1) suggest that our model surpasses the baseline models in both metrics. Successfully pinching up the soda can with only two fingers and adjusting its pose to fit into the slot requires precise control underpinned by spatial reasoning. This capability is notably absent in *Mono*, as evident by its low success rate on both aspects.

Folding: We evaluate the success rate based on the policy's ability to perform two consecutive folds without dropping the towel. The results are reported in Tab. 1. Both ours and *ResNet* models achieve a 100% success rate in performing the folds. We speculate that the high success rate of this task across all models is due to its high repetitiveness in actions. On the other hand, *Mono* occasionally fails (2/5 fails) at the third stage, in which the robot should gently move the towel to the edge of the table to ease the second fold. We observed that *Mono* tends to press its hands too hard on the table and prohibit the successful movement of the towel. This action is likely due to the lack of depth information from using a single RGB image, as this step requires the robot to adjust the force exerted by its hands based on the distance to the table.

Unloading: We evaluate the success rate on three consecutive stages individually: extraction of the tube, in-hand passing, and placing. Based on the results in Tab. 1, our model reaches 100% success rate in all three stages. For both *ResNet* and *Mono*, they fail at extracting the tube from certain slots (*ResNet* 3/20 fails, *Mono* 6/20 fails). The extraction is particularly hard for *Mono*, partly because it cannot estimate the relative orientation between the tube and the robot hand correctly. In-hand passing and placing are relatively much simpler, mostly because these two stages do not involve much randomization during data collection. Nonetheless, *ResNet* and *Mono* still fail at passing or placing occasionally (both 1/20 fails) while our model completed these two stages with no mistake.

3.2 Generalization

We conduct an experiment to evaluate the generalization capabilities of our model under randomized conditions. In the *Can Sorting* task conducted with H1, we assessed its success rate of picking from a 4x4 grid with, each cell measuring 3 cm, as depicted in Fig.5 (left). The results are detailed in Fig.5 (right), which indicate that our policy generalizes well to large areas covered in the dataset, maintaining a 100% success rate. Even in the peripheral regions, which are rare or absent in the demonstrations, our model still exhibits adaptability to complete the grasp at times.

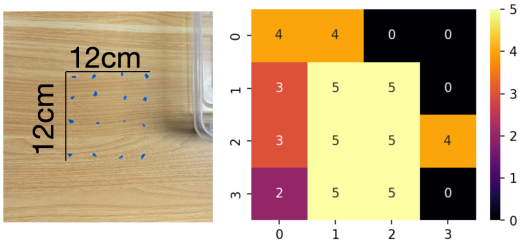


Figure 5: Left: Illustration of 4×4 sample locations of cans. Right: number of successful pickings heatmap with 5 trials at each location.

3.3 User Study

We validate our design choice of streaming and displaying stereo images in the VR headset through a user study. The use study is performed on four participants with varied levels of prior exposure to VR devices. The participants are graduate students aged 20-25. Each participant is asked to complete all four tasks under guidance and is given roughly five minutes to familiarize with the system and the tasks. Their performances in both stereo and monocular (*Mono*) setup are detailed in Tab. 2. Both metrics (task completion time and success rate) suggest that *Stereo* surpasses *Mono* by a large margin. Additionally, based on qualitative results of user feedback, using stereo image is remarkably

User	Stereo				Mono			
	Can Sorting	Can Insertion	Folding	Unloading	Can Sorting	Can Insertion	Folding	Unloading
#1	67 / 1.00	54 / 1.00	30 / 1.00	73 / 1.00	110 / 1.00	116 / 0.33	49 / 1.00	94 / 0.50
#2	59 / 1.00	55 / 1.00	44 / 1.00	87 / 1.00	91 / 1.00	92 / 0.83	67 / 1.00	119 / 0.50
#3	55 / 1.00	59 / 1.00	27 / 1.00	52 / 1.00	76 / 0.90	68 / 0.83	37 / 1.00	96 / 0.50
#4	82 / 1.00	62 / 1.00	36 / 1.00	55 / 1.00	88 / 0.80	62 / 0.83	47 / 1.00	78 / 0.50
Mean	66 / 1.00	58 / 1.00	34 / 1.00	67 / 1.00	91 / 0.93	85 / 0.71	50 / 1.00	97 / 0.50

Table 2: Each cell depicts completion time (seconds) / success rate for individual participant. Users see stereo video stream in *Stereo* and see a single RGB video from the left camera in *Mono*.

better than using a single RGB image. Typically, human eyes are adaptive enough to infer depth and spatial relationships from a single RGB image. However, in teleoperation cases where the operator must actively interact with different objects, such intuitions are often proven insufficient. Leveraging the spatial information available in stereo images can substantially alleviate the discomfort when teleoperating robots with only images.

4 Related Work

Learning based methods have achieved great success in locomotion [34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46], utilizing massively available simulation data. Collecting high-quality data with real robots for manipulation [47, 48, 1, 49, 50] has also shown its potential for training general manipulation policies.

Data collection. Recent advances in learning-based methods have given rise to the emergence of diverse data collection pipelines for manipulation, including using RGB cameras [4, 5, 6], mocap gloves [7, 8, 9], and VR devices [3, 2]. There are also more conventional frameworks exploiting exo-skeleton devices [11] or mirroring arms for joint copy [10, 12, 13]. For example, the ALOHA framework [10] provides precise control on fine-grained manipulation tasks.

Bimanual Teleoperation. Access to an intuitive and responsive teleoperation system is essential for collecting high-fidelity demonstrations. Most existing teleoperation systems are restricted to using grippers [11, 51, 13, 14] or single-hand setups [52, 53, 54], which tend to be either non-intuitive or limited in their capabilities. We believe enabling manipulation with multi-finger hands in 3D space allows more robust manipulation conducted on diverse tasks, providing significant advantages over existing teleoperation systems. For example, it will be very challenging for a parallel-jaw gripper to stably grasp a Pringles Chips tube. For the handful of bimanual-hands setups [3, 2], they require operators to control the robot by directly observing its hands during task execution or seeing an RGB image from a fixed camera. In contrast, our system integrates a first-person stereo display with active head rotations, offering the most intuitive interfaces as if the robot is an avatar of the operator itself.

Imitation learning. Imitation learning from real-robot data has been studied in [14, 10, 12, 55, 56, 57, 58, 48, 59, 1, 60, 49, 61]. Learning from play data has been studied in [62, 63, 64]. Transferring human data to robot policies has been studied in [63, 65, 66, 67, 68, 69, 70, 54]. Imitating motions for physical characters and real robots is studied in [71, 72, 38, 73, 74, 75, 76, 77, 78, 76, 79, 80, 81, 82, 83, 84].

5 Conclusion and Limitations

In this paper, we propose a system for immersive teleoperation with stereoscopic video streaming and active perception with actuated necks. We show that our system can enable precise and long-horizon manipulation tasks and the data collected with our system can be readily utilized by imitation learning algorithms. We also conduct user studies to show the importance of stereo perception for human operators. While the stereo perception is crucial for the operators' spatial understanding, there is still a lack of other forms of feedback, such as haptic feedback, which is typically the dominant feedback with first-person visual occlusion and in tactile-intensive tasks. A system that enables the relabeling of expert data could be very helpful for increasing success rate, which is also missing from our system now. The future work can be extended to mobile version which utilizes all the DoFs of the robot.

References

[1] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[2] A. Iyer, Z. Peng, Y. Dai, I. Guzey, S. Haldar, S. Chintala, and L. Pinto. Open teach: A versatile teleoperation system for robotic manipulation, 2024.

[3] T. Lin, Y. Zhang, Q. Li, H. Qi, B. Yi, S. Levine, and J. Malik. Learning visuotactile skills with two multifingered hands. *arXiv:2404.16823*, 2024.

[4] 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. In *Robotics: Science and Systems*, 2023.

[5] A. Sivakumar, K. Shaw, and D. Pathak. Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube. *arXiv preprint arXiv:2202.10448*, 2022.

[6] S. Li, J. Jiang, P. Ruppel, H. Liang, X. Ma, N. Hendrich, F. Sun, and J. Zhang. A mobile robot hand-arm teleoperation system by vision and imu. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10900–10906. IEEE, 2020.

[7] M. Caeiro-Rodríguez, I. Otero-González, F. A. Mikic-Fonte, and M. Llamas-Nistal. A systematic review of commercial smart gloves: Current status and applications. *Sensors*, 21(8):2667, 2021.

[8] H. Liu, Z. Zhang, X. Xie, Y. Zhu, Y. Liu, Y. Wang, and S.-C. Zhu. High-fidelity grasping in virtual reality using a glove-based system. In *2019 international conference on robotics and automation (icra)*, pages 5180–5186. IEEE, 2019.

[9] H. Liu, X. Xie, M. Millar, M. Edmonds, F. Gao, Y. Zhu, V. J. Santos, B. Rothrock, and S.-C. Zhu. A glove-based system for studying hand-object manipulation via joint pose and force sensing. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6617–6624. IEEE, 2017.

[10] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[11] H. Fang, H.-S. Fang, Y. Wang, J. Ren, J. Chen, R. Zhang, W. Wang, and C. Lu. Low-cost exoskeletons for learning whole-arm manipulation in the wild. *arXiv preprint arXiv:2309.14975*, 2023.

[12] L. X. Shi, Z. Hu, T. Z. Zhao, A. Sharma, K. Pertsch, J. Luo, S. Levine, and C. Finn. Yell at your robot: Improving on-the-fly from language corrections. *arXiv preprint arXiv: 2403.12910*, 2024.

[13] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel. Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators. *arXiv preprint arXiv:2309.13037*, 2023.

[14] Z. Fu, T. Z. Zhao, and C. Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.

[15] C. Wang, H. Shi, W. Wang, R. Zhang, L. Fei-Fei, and C. K. Liu. Dexcap: Scalable and portable mocap data collection system for dexterous manipulation. *arXiv preprint arXiv:2403.07788*, 2024.

[16] Y. Park and P. Agrawal. Using apple vision pro to train and control robots, 2024. URL https://github.com/Improbable-AI/VisionProTeleop.

[17] Unitree Robotics, H1, 2024, www.unitree.com/h1, [Online; accessed Feb. 2024].

[18] Fourier Intelligence, GR-1, 2024, www.fourierintelligence.com/gr1, [Online; accessed Jun. 2024].

[19] Inspire Robots, Dexterous Hands, 2024, www.inspire-robots.store/collections/the-dexterous-hands, [Online; accessed Jun. 2024].

[20] Vuer, 2024, docs.vuer.ai/en/latest, [Online; accessed Jun. 2024].

[21] Dynamixel-X, 2024, www.robotis.us/dynamixel-xl330-m288-t, [Online; accessed Jun. 2024].

[22] Zed Mini, Stereo Camera, 2024, www.robotis.us/dynamixel-xl330-m288-t, [Online, accessed Jun. 2024].

[23] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard. The pinocchio c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives. In *IEEE International Symposium on System Integrations (SII)*, 2019.

[24] J. Carpentier, F. Valenza, N. Mansard, et al. Pinocchio: fast forward and inverse dynamics for poly-articulated systems. https://stack-of-tasks.github.io/pinocchio, 2015–2021.

[25] Pinocchio CLIK, 2024, gepettoweb.laas.fr/doc/stack-of-tasks/pinocchio/master/doxygen-html/md_doc_b-examples_i-inverse-kinematics.html, [Online, accessed Jun. 2024].

[26] 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)*, pages 9164–9170. IEEE, 2020.

[27] D. Kraft. Algorithm 733: TOMP–fortran modules for optimal control calculations. *ACM Transactions on Mathematical Software*, 20:262–281, 1994. doi:10.1145/192115.192124.

[28] S. G. Johnson. The NLopt nonlinear-optimization package. https://github.com/stevengj/nlopt, 2007.

[29] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[30] T. Darcet, M. Oquab, J. Mairal, and P. Bojanowski. Vision transformers need registers, 2023.

[31] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi:10.1109/CVPR.2016.90.

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

[33] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://arxiv.org/abs/1711.05101.

[34] G. B. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal. Rapid locomotion via reinforcement learning. *arXiv preprint arXiv:2205.02824*, 2022.

[35] A. Kumar, Z. Fu, D. Pathak, and J. Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.

[36] Z. Fu, A. Kumar, J. Malik, and D. Pathak. Minimizing energy consumption leads to the emergence of gaits in legged robots. *Conference on Robot Learning (CoRL)*, 2021.

[37] J. Fu, Y. Song, Y. Wu, F. Yu, and D. Scaramuzza. Learning deep sensorimotor policies for vision-based autonomous drone racing, 2022.

[38] A. Escontrela, X. B. Peng, W. Yu, T. Zhang, A. Iscen, K. Goldberg, and P. Abbeel. Adversarial motion priors make good substitutes for complex reward functions. 2022 ieee. In *International Conference on Intelligent Robots and Systems (IROS)*, volume 2, 2022.

[39] Z. Li, X. Cheng, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath. Reinforcement learning for robust parameterized locomotion control of bipedal robots. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2811–2817. IEEE, 2021.

[40] J. Siekmann, K. Green, J. Warila, A. Fern, and J. Hurst. Blind bipedal stair traversal via sim-to-real reinforcement learning. *arXiv preprint arXiv:2105.08328*, 2021.

[41] A. Agarwal, A. Kumar, J. Malik, and D. Pathak. Legged locomotion in challenging terrains using egocentric vision. In *Conference on Robot Learning*, pages 403–415. PMLR, 2023.

[42] X. Cheng, A. Kumar, and D. Pathak. Legs as manipulator: Pushing quadrupedal agility beyond locomotion. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[43] Z. Fu, X. Cheng, and D. Pathak. Deep whole-body control: learning a unified policy for manipulation and locomotion. In *Conference on Robot Learning*, pages 138–149. PMLR, 2023.

[44] H. Duan, B. Pandit, M. S. Gadde, B. J. van Marum, J. Dao, C. Kim, and A. Fern. Learning vision-based bipedal locomotion for challenging terrain. *arXiv preprint arXiv:2309.14594*, 2023.

[45] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao. Robot parkour learning. In *Conference on Robot Learning (CoRL)*, 2023.

[46] X. Cheng, K. Shi, A. Agarwal, and D. Pathak. Extreme parkour with legged robots. *arXiv preprint arXiv:2309.14341*, 2023.

[47] 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*, 31, 2018.

[48] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, P. D. Fagan, J. Hejna, M. Itkina, M. Lepert, Y. J. Ma, P. T. Miller, J. Wu, S. Belkhale, S. Dass, H. Ha, A. Jain, A. Lee, Y. Lee, M. Memmel, S. Park, I. Radosavovic, K. Wang, A. Zhan, K. Black, C. Chi, K. B. Hatch, S. Lin, J. Lu, J. Mercat, A. Rehman, P. R. Sanketi, A. Sharma, C. Simpson, Q. Vuong, H. R. Walke, B. Wulfe, T. Xiao, J. H. Yang, A. Yavary, T. Z. Zhao, C. Agia, R. Baijal, M. G. Castro, D. Chen, Q. Chen, T. Chung, J. Drake, E. P. Foster, J. Gao, D. A. Herrera, M. Heo, K. Hsu, J. Hu, D. Jackson, C. Le, Y. Li, K. Lin, R. Lin, Z. Ma, A. Maddukuri, S. Mirchandani, D. Morton, T. Nguyen, A. O'Neill, R. Scalise, D. Seale, V. Son, S. Tian, E. Tran, A. E. Wang, Y. Wu, A. Xie, J. Yang, P. Yin, Y. Zhang, O. Bastani, G. Berseth, J. Bohg, K. Goldberg, A. Gupta, A. Gupta, D. Jayaraman, J. J. Lim, J. Malik, R. Martín-Martín, S. Ramamoorthy, D. Sadigh, S. Song, J. Wu, M. C. Yip, Y. Zhu, T. Kollar, S. Levine, and C. Finn. Droid: A large-scale in-the-wild robot manipulation dataset. 2024.

[49] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

[50] N. M. M. Shafiullah, A. Rai, H. Etukuru, Y. Liu, I. Misra, S. Chintala, and L. Pinto. On bringing robots home. *arXiv preprint arXiv:2311.16098*, 2023.

[51] M. Seo, S. Han, K. Sim, S. H. Bang, C. Gonzalez, L. Sentis, and Y. Zhu. Deep imitation learning for humanoid loco-manipulation through human teleoperation. In *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, pages 1–8. IEEE, 2023.

[52] S. P. Arunachalam, I. Güzey, S. Chintala, and L. Pinto. Holo-dex: Teaching dexterity with immersive mixed reality. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5962–5969. IEEE, 2023.

[53] Y. Qin, H. Su, and X. Wang. From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation. *IEEE Robotics and Automation Letters*, 7(4): 10873–10881, 2022.

[54] J. Wang, Y. Qin, K. Kuang, Y. Korkmaz, A. Gurumoorthy, H. Su, and X. Wang. CyberDemo: Augmenting Simulated Human Demonstration for Real-World Dexterous Manipulation. *arXiv preprint arXiv: 2312.09237*, 2024.

[55] 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 Robotics: Science and Systems (RSS)*, 2024.

[56] 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.

[57] J. Pari, N. M. Shafiullah, S. P. Arunachalam, and L. Pinto. The surprising effectiveness of representation learning for visual imitation. *arXiv preprint arXiv:2112.01511*, 2021.

[58] 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 *arXiv preprint arXiv:2108.03298*, 2021.

[59] A. Padalkar, A. Pooley, A. Jain, A. Bewley, A. Herzog, A. Irpan, A. Khazatsky, A. Rai, A. Singh, A. Brohan, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

[60] M. Shridhar, L. Manuelli, and D. Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pages 785–799. PMLR, 2023.

[61] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[62] Z. J. Cui, Y. Wang, N. M. M. Shafiullah, and L. Pinto. From play to policy: Conditional behavior generation from uncurated robot data. *arXiv preprint arXiv:2210.10047*, 2022.

[63] 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.

[64] R. Mendonca, S. Bahl, and D. Pathak. Alan : Autonomously exploring robotic agents in the real world. *ICRA*, 2023.

[65] H. Xiong, R. Mendonca, K. Shaw, and D. Pathak. Adaptive mobile manipulation for articulated objects in the open world. *arXiv preprint arXiv:2401.14403*, 2024.

[66] S. Bahl, A. Gupta, and D. Pathak. Human-to-robot imitation in the wild. *arXiv preprint arXiv:2207.09450*, 2022.

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

[68] 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*, pages 18995–19012, 2022.

[69] Y. Zhu, A. Lim, P. Stone, and Y. Zhu. Vision-based manipulation from single human video with open-world object graphs. *arXiv preprint arXiv:2405.20321*, 2024.

[70] R. Mendonca, S. Bahl, and D. Pathak. Structured world models from human videos. *RSS*, 2023.

[71] X. B. Peng, E. Coumans, T. Zhang, T.-W. Lee, J. Tan, and S. Levine. Learning agile robotic locomotion skills by imitating animals. Apr. 2020.

[72] Y. Wang, Z. Jiang, and J. Chen. Amp in the wild: Learning robust, agile, natural legged locomotion skills. *arXiv preprint arXiv:2304.10888*, 2023.

[73] Y. Fuchioka, Z. Xie, and M. Van de Panne. Opt-mimic: Imitation of optimized trajectories for dynamic quadruped behaviors. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5092–5098. IEEE, 2023.

[74] R. Yang, Z. Chen, J. Ma, C. Zheng, Y. Chen, Q. Nguyen, and X. Wang. Generalized animal imitator: Agile locomotion with versatile motion prior. *arXiv preprint arXiv:2310.01408*, 2023.

[75] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics (ToG)*, 40(4):1–20, 2021.

[76] X. B. Peng, Y. Guo, L. Halper, S. Levine, and S. Fidler. Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Trans. Graph.*, 41(4), July 2022.

[77] C. Tessler, Y. Kasten, Y. Guo, S. Mannor, G. Chechik, and X. B. Peng. Calm: Conditional adversarial latent models for directable virtual characters. In *ACM SIGGRAPH 2023 Conference Proceedings*, SIGGRAPH '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701597. doi:10.1145/3588432.3591541. URL https://doi.org/10.1145/3588432.3591541.

[78] M. Hassan, Y. Guo, T. Wang, M. Black, S. Fidler, and X. B. Peng. Synthesizing physical character-scene interactions. 2023. doi:10.1145/3588432.3591525. URL https://doi.org/10.1145/3588432.3591525.

[79] J. Won, D. Gopinath, and J. Hodgins. A scalable approach to control diverse behaviors for physically simulated characters. *ACM Trans. Graph.*, 39(4), 2020. URL https://doi.org/10.1145/3386569.3392381.

[80] T. Wang, Y. Guo, M. Shugrina, and S. Fidler. Unicon: Universal neural controller for physics-based character motion, 2020.

[81] H. Zhang, Y. Yuan, V. Makoviychuk, Y. Guo, S. Fidler, X. B. Peng, and K. Fatahalian. Learning physically simulated tennis skills from broadcast videos. *ACM Trans. Graph.*, 42(4), jul 2023. ISSN 0730-0301. doi:10.1145/3592408. URL https://doi.org/10.1145/3592408.

[82] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Trans. Graph.*, 37(4):143:1–143:14, July 2018. ISSN 0730-0301. doi:10.1145/3197517.3201311. URL http://doi.acm.org/10.1145/3197517.3201311.

[83] X. Cheng, Y. Ji, J. Chen, R. Yang, G. Yang, and X. Wang. Expressive whole-body control for humanoid robots. *arXiv preprint arXiv:2402.16796*, 2024.

[84] T. He, Z. Luo, W. Xiao, C. Zhang, K. Kitani, C. Liu, and G. Shi. Learning human-to-humanoid real-time whole-body teleoperation. In *arXiv*, 2024.