

Supplementary Materials for exUMI: Extensible Robot Teaching System with Action-aware Task-agnostic Tactile Representation

Anonymous Author(s)

Affiliation

Address

email

1 Details of exUMI Hardware Design

We gave a brief introduction to the hardware and algorithms for exUMI due to the page limit. Below are the details of our system.

1.1 Hardware

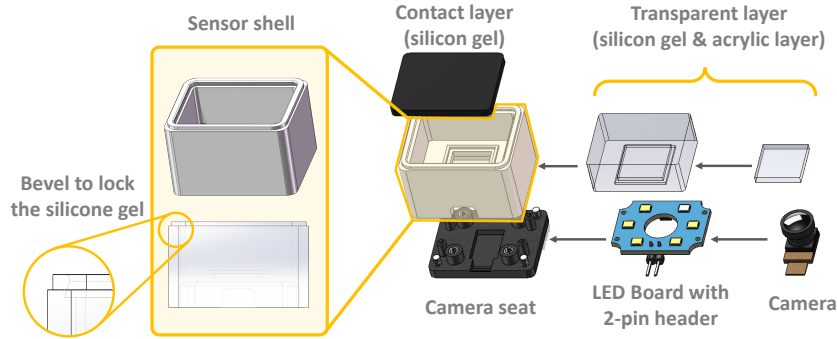


Figure 1: An exploded view of the enhanced tactile sensor design. We enhance the 9DTact for stability and quality control. We add an inverted bevel to the sensor shell to secure the black silicon gel and prevent it from detaching.

Magnetic Rotary Encoder. We propose a low-cost AS5600 magnetic rotary encoder solution to achieve accurate and robust gripper state capture. AS5600 is a contactless rotary position encoder that utilizes Hall effect sensing to measure angular position. As shown in Fig. 3, we modify the top cover of UMI to attach a radial magnet to one of the joints of the mechanical assembly, with the Hall sensor positioned above it at an appropriate distance (~ 2 mm). The AS5600 provides high-resolution 12-bit position readings (4,096 positions per revolution) and communicates with the single-board computer through the I²C protocol. This solution offers a higher sampling rate and resolution, immunity to visual occlusion, and negligible computational overhead.

AR Motion Capture System. To overcome the limitations of vision-based tracking (SLAM) in occluded or complex scenarios, we adopt an AR-based approach for end-effector pose estimation. Following ARCap [1], our system uses a Meta Quest 3 headset for 6D pose motion capture, which is accurate and robust to occlusion. We integrate the left VR controller through a custom-designed mount attached to the UMI body, and use the headset to track the 6D pose of the controller. The mount also provides additional space for the power supply and an Orange Pi controller, serving as a universal sensor hub, synchronously capturing data from the AR headset, rotary encoder, and any additional sensors, such as tactile sensors.

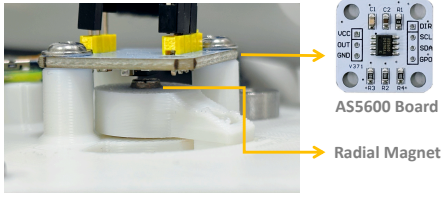


Figure 3: Detailed view of AS5600 sensor on exUMI.

Component	Base Cost (\$)
GoPro 11 + Accessories	298
Meta Quest VR Headset	299
Orange Pi 3B /	35
AS5600 Magnetic Encoder	1
3D Printed Parts	15
Visuo-Tactile Sensors	30
Misc. (Power Bank, Cables/Screws/Nuts)	20
Total Cost of exUMI	698

Table 1: Bill of materials (BOM) of exUMI.

The orientation of the VR controller is arbitrary since the transformation between the controller and UMI coordinate frames will be determined through our calibration pipeline. This flexible mounting approach simplifies the assembly and avoids precise physical alignment.

Visual Input. Same as the UMI [2], we employ a GoPro camera with a fisheye lens as our primary visual input. We move the camera forward following FastUMI [3] for a wider and clearer view. The camera positioning eliminates body occlusion in the field of view to enhance transferability.

Fingertip Visuo-tactile Sensors. We improve 9D-Tact [4] as a low-cost and DIY-friendly tactile sensor for exUMI. As shown in Fig. 1 and Fig. 2, our enhancements involve: (1) We redesigned the sensor shell to securely anchor the top silicone layer, enhancing its resistance to tangential forces and ensuring long-term stability. (2) The LED board was modified to incorporate a more robust 2-pin header connector for stable power delivery. Compared to USB cables, Dupont connectors offer superior cable management flexibility. Plus, the LEDs were rearranged to minimize power consumption—a critical factor for our embedded system’s efficiency. (3) A custom mold was developed to precisely control the silicone layer’s thickness. The mold is affixed to the sensor shell, allowing controlled pouring of transparent, translucent, or opaque liquid silicone until it reaches the desired level (Fig. 2). Excess silicone is then removed by carefully scraping along the mold’s surface with a spatula. The upgraded sensor design achieves significantly enhanced durability and stability. Please refer to the supplementary for more fabrication details.

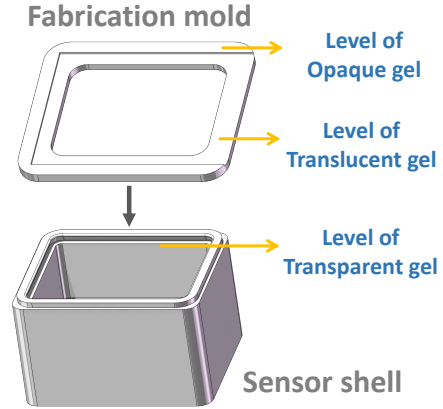


Figure 2: The mold for stable fabrication of the tactile sensor.

Cost and Accessibility. We show the overall bill of materials in Tab. 1. Our system is low-cost with a minimal configuration starting at \$ 698, which can be further reduced by substituting the GoPro with alternative fisheye cameras. Our design emphasizes DIY-friendly assembly and readily available components, making it suitable for research/education. All design files will be released.

1.2 Data Collection and Processing

AR Capture Interface. Building upon the remarkable engineering of ARCap [1], we simplify the socket-based data transfer interface for the 6D pose capture process. The collection procedure is as follows:

1. Initialize the server program on the Raspberry Pi.
2. Launch the client application on the Meta Quest headset.
3. Set up the base coordinate frame in AR space.
4. Begin data streaming of real-time 6D controller poses to the Raspberry Pi.

Algorithm 1 Latency alignment algorithm

Input: Trajectories $f(t)$ and $g(t)$, timesteps $\{t_i\}_{i=1}^T$, bounds of latency δ_{min} δ_{max}

Input: Constants: $\epsilon = 0.0001$, search window N , search splits M

Output: Latency δ^* of $g(t)$ such that $f(t) \approx g(t + \delta^*)$

- 1: **repeat**
 - 2: Interpolate the interval $[\delta_{min}, \delta_{max}]$ into M segments: $\delta_0, \delta_1, \dots, \delta_M$
 - 3: $k = \min_k \sum_{i=1}^T \|f(t_i) - g(t_i + \delta_k)\|_2^2$
 - 4: $\delta^* = \delta_k$
 - 5: $\delta_{min}, \delta_{max} \leftarrow \delta_{k-N}, \delta_{k+N}$ (update the search range to the neighborhood of δ_k)
 - 6: **until** $\delta_{max} - \delta_{min} < \epsilon$
-

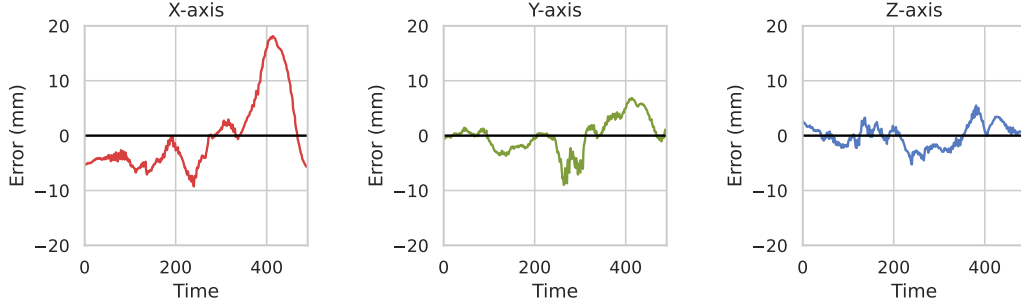


Figure 4: Comparison of AR MoCap trajectory and ground truth trajectory.

59 **Calibration of AR Latency.** To synchronize AR motion capture with the visual inputs, we designed
60 a calibration protocol involving horizontal sweeps in front of a stationary ArUco marker. We extract
61 the x-axis movement of the AR MoCap system and the camera-detected marker trajectories, then use
62 a bisection-style optimization algorithm to align the two trajectories and compute the latency of the
63 AR system. Specifically, given the two 1-dimension trajectories on timesteps $\{t_i\}_{i=1}^T$, we convert
64 them to two function $f(t)$ $g(t)$ w.r.t time t by interpolation, which is then calibrated by minimizing
65 the MSE error between the two trajectory. The details are given in Alg. 1.

66 **Data Collection and Processing Pipeline.** Compared to the original UMI system, our data collec-
67 tion and processing pipeline is significantly simplified and more robust:

- 68 1. Set up the desired environment.
- 69 2. Initialize AR tracking system on the Raspberry Pi.
- 70 3. Record latency calibration sequence (one video).
- 71 4. Record demonstration videos.
- 72 5. Calculate and apply temporal latency correction.
- 73 6. Align AR capture data to the video frames through interpolation.
- 74 7. Pack synchronized data.

75 1.3 System Evaluation

76 **Proprioception Precision.** We first evaluated the precision of our proprioception system, particu-
77 larly for the AR-based MoCap system. We obtained both ground truth and AR controller trajectories
78 by mounting the AR controller on the robot end-effector and teleoperating the robot. We evaluate
79 the system by moving the robot within a 50 cm range. The resulting 6D pose differences between
80 the estimated and ground truth values are presented in Fig. 4 The system demonstrates remarkable
81 accuracy, achieving mean position errors of 5.4 / 2.3 / 1.7 mm at each axis. The rotation errors are
82 below 1 degree (notably small due to the robot’s limited rotation range). The x-axis error reaches a
83 maximum of 20 mm since it is the depth axis in the Flexiv coordinate system and inherently presents

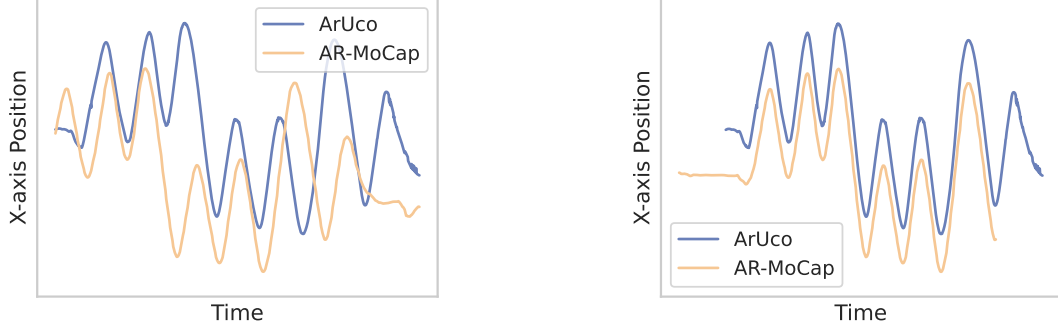


Figure 5: An example of AR MoCap and Vision trajectory before (left) and after (right) alignment.

greater measurement challenges. These high-precision measurements enable efficient robot policy learning by providing high-quality training data.

Latency Calibration. We give a qualitative visualization of our latency calibration algorithm in Fig. 5. The x-axis trajectory of the AR MoCap system and the visual input (represented as the trajectory of the ArUco marker) are perfectly aligned after our calibration system, and we can read the time offset for further modality alignment. With proper sweeping frequency, our system could consistently achieve less than 5 ms latency error.

Key Takeaways

We provide an accurate, extensible, cost-effective, and DIY-friendly enhancement to the UMI system. Our solution is particularly valuable if you need:

- **Enhanced Tracking Precision:** Achieve $\sim 100\%$ data effective ratio in diverse environments by our AR-based MoCap system and the magnetic rotary encoder for gripper state.
- **Non-Parallel Gripper Support:** An alternative design is provided for more popular industrial grippers like Flexiv Grav or Robotiq 2F series with an adaptable mechanical design.
- **Multimodal Sensing:** Including tactile, audio, or other custom sensors through our modular hardware interface.

All components are commercially available, and the fabrication details will be opened soon.

2 Detailed Taxonomy of Tactile Representation Learning

We give more details of our discussion on the current taxonomy of tactile representation learning.

The target of tactile representation learning is to learn a tactile encoder \mathcal{E}_T for the tactile data, to facilitate further multimodal policy learning:

$$\pi(\mathbf{a}_{t+1} | \mathcal{E}_S(\mathbf{s}_t), \mathcal{E}_T(\mathbf{T}_t), \mathcal{E}_V(\mathbf{V}_t)). \quad (1)$$

Current tactile representation methods fall into three dominant paradigms, each with specific advantages and fundamental constraints:

(a) Direct Multimodal Imitation Learning [5, 6, 7, 8, 9]. This approach trains end-to-end multimodal policies in Eq. 1 using paired tactile-visual-action data, and learn the tactile representation ϕ_T . While effective for narrow tasks, it suffers from tactile data scarcity since, in most regular robot tasks, the tactile contacts only occur in very few frames. The method also inherently couples task objectives with tactile features, limiting cross-task transferability.

(b) Spatial Self-Supervised Learning [10, 11, 12, 13]. Self-supervised learning (SSL) is adopted for a generic and transferable tactile representation, and it has been widely adopted for frame-level tactile pretraining. Methods like contrastive learning [10, 11, 12] and masked learning [13] learn tactile embeddings $\mathcal{E}_T(\mathbf{T}_t)$ through proxy objectives on task-agnostic unlabeled data. While reducing human annotation costs, SSL usually imposes incorrect inductive biases borrowed from vision:

Dataset	Data Scale	Tactile Sensor	Proprioception / Action	Collection Source
Calandra et al. [16]	6.5 K	GelSight	✓	Robot
Calandra et al. [17]	9.3 K	GelSight	✓	Robot
VisGel [14]	12.0 K	GelSight	✓	Robot
Burka [18]	1.1 K	Multiple	✓	Human
Touch and Go [19]	13.9 K	GelSight	✗	Human
ObjectFolder Real [20]	3.0 K	GelSight	✓	Robot
SSVTP [21]	4.5 K	DIGIT	✓	Robot
TVL [22]	43.7 K	DIGIT	✓	Robot
Touch2Touch [23]	32.3 K	Multiple	✓	Robot
X-Capture [15]	3.0 K	DIGIT	✗	Human
Ours	480.9 K (raw frames)	9DTact+	✓	Human

Table 2: Comparison of real-world tactile datasets.

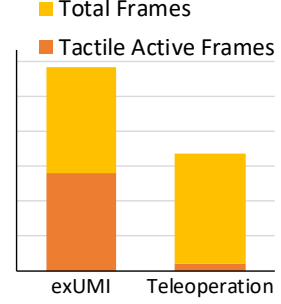


Figure 6: Comparison of per-hour tactile data collection efficiency.

for instance, masked learning assumes geometrical self-consistency, but tactile signals could be visually arbitrary; contrastive learning usually assumes translation invariance, which may not exist in tactile sensing. It is challenging to design a self-supervising proxy objective for a robot task.

(c) Visual-Tactile Alignment [14, 15]. Cross-modal alignment learns joint embeddings by maximizing in-pair similarity $s(\mathcal{E}_T(\mathbf{T}_t), \mathcal{E}_V(\mathbf{V}_t))$ of visual and tactile modalities. Though it is effective for visual-language learning, it fundamentally assumes a coarse *one-to-one visuo-tactile mapping*, regardless of the actual *one-to-many relation*: with different contact forces, identical visual scenes yield divergent tactile signals. The multimodal alignment also overlooks that visual and tactile sensing are complementary rather than well-aligned. For robot learning, tactile sensors are a complementary information to the visual input, but the alignment method discards this privileged information.

Currently, these pretraining approaches face limitations that stem from a shared oversight: treating tactile signals as static observations rather than *action-aware dynamic processes*. Hence, our framework bridges this gap by reformulating tactile representation as an action-conditioned temporal prediction problem, explicitly modeling the forward tactile dynamics that underpin real-world contact interactions.

3 More Implementation Details

Tactile Data Curation. Since active tactile signals are sparse in real-world data collection, we adopt a data rejection strategy during data sampling to avoid trivial samples. For each data chunk, we check the proportion of active pixels for each tactile frame. If the active proportion of all frames is below a certain threshold, the data chunk will be discarded and resampled.

Implementation Details. For each timestep, our exUMI collects two tactile images on the two sides of the gripper. We convert the images to a calibrated grayscale image following 9DTact [4], and extract the convex and concave pixel map by comparing the grayscale image to the reference image (tactile signal at no contact). The grayscale image, convex map, and concave map are stacked as a 3-channel image for a richer representation of tactile contacts.

We pretrain the tactile representation on our large-scale human play dataset, which is randomly split into a training and validation set by 15:1. The action sequence is represented as the relative pose and the gripper state. The images on two sides are concatenated and then downsampled to 224×224 resolution. We use a pretrained VAE model (KL-F16) as the encoder and decoder for tactile learning. The tactile prediction is conducted in 8 temporal frames, where 4 random frames in the first half are regarded as input and 4 frames in the second half are the prediction target. We adopt a larger frequency for action following Li et al. [24]. The tactile prediction model reaches quick convergence due to the simpler distribution of tactile sensing.

Environment Details. The camera and sensors are connected to a master computer with a

142 RTX 4070 GPU, which controls the robot at a fre-
143 quency of 10 Hz. As illustrated in Fig. 7, we designed
144 a pipe clamp-style camera mount and gripper tactile
145 sensor mount to exactly replicate the end-effector sen-
146 sor placement of the exUMI system. The mount con-
147 sists of two half-rings, one integrated with a standard
148 GoPro mount. This design can be attached to the
149 Flexiv Grav gripper base, and also adaptable on var-
150 ious other end effectors.

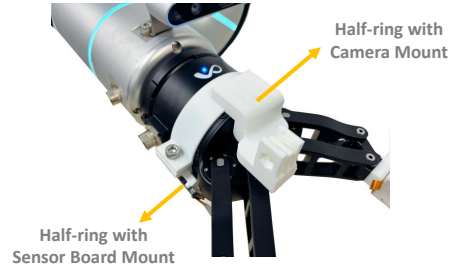


Figure 7: Pipe clamp style GoPro mount for deployment.

References

- [1] 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.
- [2] 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. *arXiv preprint arXiv:2402.10329*, 2024.
- [3] Z. Wu, T. Wang, C. Guan, Z. Jia, S. Liang, H. Song, D. Qu, D. Wang, Z. Wang, N. Cao, et al. Fast-umi: A scalable and hardware-independent universal manipulation interface. *arXiv preprint arXiv:2409.19499*, 2024.
- [4] C. Lin, H. Zhang, J. Xu, L. Wu, and H. Xu. 9dtact: A compact vision-based tactile sensor for accurate 3d shape reconstruction and generalizable 6d force estimation. *IEEE Robotics and Automation Letters*, 2023.
- [5] E. Su, C. Jia, Y. Qin, W. Zhou, A. Macaluso, B. Huang, and X. Wang. Sim2real manipulation on unknown objects with tactile-based reinforcement learning. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9234–9241, 2024. doi:10.1109/ICRA57147.2024.10611113.
- [6] K.-W. Lee, Y. Qin, X. Wang, and S.-C. Lim. Dextouch: Learning to seek and manipulate objects with tactile dexterity. *IEEE Robotics and Automation Letters*, 9(12):10772–10779, Dec. 2024. ISSN 2377-3774. doi:10.1109/lra.2024.3478571. URL <http://dx.doi.org/10.1109/LRA.2024.3478571>.
- [7] B. Romero, H.-S. Fang, P. Agrawal, and E. Adelson. Eyesight hand: Design of a fully-actuated dexterous robot hand with integrated vision-based tactile sensors and compliant actuation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1853–1860, 2024. doi:10.1109/IROS58592.2024.10802778.
- [8] H. Lin, R. Corcoran, and D. Zhao. Generalize by touching: Tactile ensemble skill transfer for robotic furniture assembly. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9227–9233, 2024. doi:10.1109/ICRA57147.2024.10610567.
- [9] V. Pattabiraman, Y. Cao, S. Haldar, L. Pinto, and R. Bhirangi. Learning precise, contact-rich manipulation through uncalibrated tactile skins, 2024. URL <https://arxiv.org/abs/2410.17246>.
- [10] S. Rodriguez, Y. Dou, W. van den Bogert, M. Oller, K. So, A. Owens, and N. Fazeli. Contrastive touch-to-touch pretraining, 2024. URL <https://arxiv.org/abs/2410.11834>.
- [11] I. Guzey, B. Evans, S. Chintala, and L. Pinto. Dexterity from touch: Self-supervised pre-training of tactile representations with robotic play, 2023. URL <https://arxiv.org/abs/2303.12076>.
- [12] K. Yu, Y. Han, Q. Wang, V. Saxena, D. Xu, and Y. Zhao. Mimictouch: Leveraging multi-modal human tactile demonstrations for contact-rich manipulation, 2025. URL <https://arxiv.org/abs/2310.16917>.
- [13] T. Wu, J. Li, J. Zhang, M. Wu, and H. Dong. Canonical representation and force-based pretraining of 3d tactile for dexterous visuo-tactile policy learning, 2025. URL <https://arxiv.org/abs/2409.17549>.
- [14] Y. Li, J.-Y. Zhu, R. Tedrake, and A. Torralba. Connecting touch and vision via cross-modal prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10609–10618, 2019.

- 196 [15] S. Clarke, S. Wistreich, Y. Ze, and J. Wu. X-capture: An open-source portable device for
197 multi-sensory learning. *arXiv preprint arXiv:2504.02318*, 2025.
- 198 [16] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine.
199 More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and*
200 *Automation Letters*, 3(4):3300–3307, 2018.
- 201 [17] R. Calandra, A. Owens, M. Upadhyaya, W. Yuan, J. Lin, E. H. Adelson, and S. Levine.
202 The feeling of success: Does touch sensing help predict grasp outcomes? *arXiv preprint*
203 *arXiv:1710.05512*, 2017.
- 204 [18] A. L. Burka. *Instrumentation, data, and algorithms for visually understanding haptic surface*
205 *properties*. PhD thesis, University of Pennsylvania, 2018.
- 206 [19] F. Yang, C. Ma, J. Zhang, J. Zhu, W. Yuan, and A. Owens. Touch and go: Learning from
207 human-collected vision and touch, 2022. URL <https://arxiv.org/abs/2211.12498>.
- 208 [20] R. Gao, Y. Dou, H. Li, T. Agarwal, J. Bohg, Y. Li, L. Fei-Fei, and J. Wu. The object-
209 folder benchmark: Multisensory learning with neural and real objects. In *Proceedings of*
210 *the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17276–17286,
211 2023.
- 212 [21] J. Kerr, H. Huang, A. Wilcox, R. Hoque, J. Ichnowski, R. Calandra, and K. Goldberg. Self-
213 supervised visuo-tactile pretraining to locate and follow garment features. *arXiv preprint*
214 *arXiv:2209.13042*, 2022.
- 215 [22] L. Fu, G. Datta, H. Huang, W. C.-H. Panitch, J. Drake, J. Ortiz, M. Mukadam, M. Lambeta,
216 R. Calandra, and K. Goldberg. A touch, vision, and language dataset for multimodal alignment.
217 *arXiv preprint arXiv:2402.13232*, 2024.
- 218 [23] S. Rodriguez, Y. Dou, M. Oller, A. Owens, and N. Fazeli. Touch2touch: Cross-modal tactile
219 generation for object manipulation. *arXiv preprint arXiv:2409.08269*, 2024.
- 220 [24] S. Li, Y. Gao, D. Sadigh, and S. Song. Unified video action model. *arXiv preprint*
221 *arXiv:2503.00200*, 2025.