

# Zero-shot Sim2Real Transfer for Magnet-Based Tactile Sensor on Insertion Tasks

Anonymous Author(s)

Affiliation

Address

email

**Abstract:** We aim to bridge the sim-to-real gap of dense, distributed, 3-axes, magnet-based tactile sensors, like u-skin, which balances well between tactile density, high-durability, and compactness. Unlike prior works that use binarization techniques, we propose GCS, a novel sim-to-real technique to learn contact-rich insertion skills. We evaluated our approach in *blind* insertion tasks and show zero-shot sim-to-real transfer of RL policies with raw tactile readings as input.

**Keywords:** Tactile Sensor, Sim-to-Real, Insertion

## 1 Introduction

Sense of touch is an important sensing modality for humans. With dense, distributed sensors on our skin, humans can detect subtle changes in external contact, and conduct dexterous skills like tool use. There has been tremendous effort [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17] to replicate the capability on robots for manipulation. Among various tactile sensors, magnet-based sensors, like uskin [12, 13], can provide dense, distributed, 3-axes forces, and are durable and compact to be integrated into dexterous hands [18]. However, using magnet-based tactile sensors in robot learning has not been widely explored in previous work.

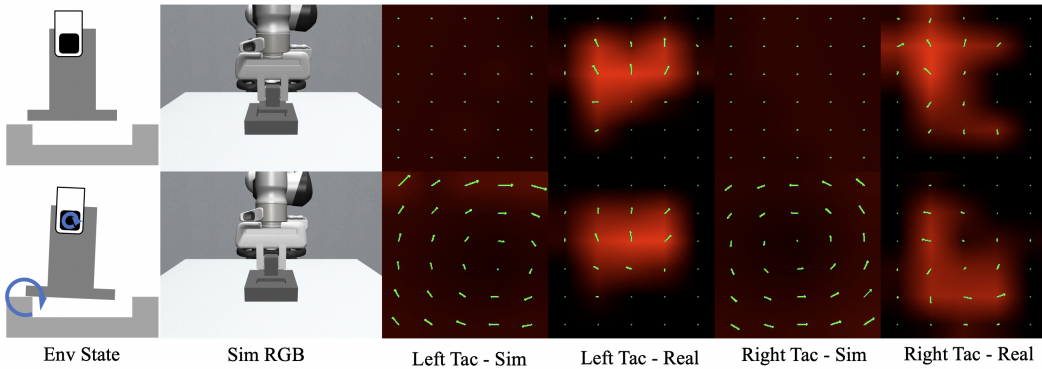


Figure 1: Comparison of simulated  $6 \times 5$  tactile readings in MuJoCo and in the real world under the same physical state. For the right four figures, the redness represents normal forces and the green arrows represent the shear forces, on each taxel. This visualizes the sim-to-real gap of tactile readings.

Our work focuses on bridging the sim-to-real gap of dense, distributed, 3-axes, magnet-based sensors [12, 19]. We use a commercial magnet-based sensor ([19]) with  $6 \times 5$  grid of  $4.7mm \times 4.7mm$  taxels, i.e.,  $6 \times 5 \times 3$  tensor readings, for *blind* insertion tasks. Most importantly, we managed to solve the task with a tactile-based RL-policy trained *only* in simulation. Furthermore, in our method, we avoid using techniques like binarization [10, 9], that loses much information from tactile sensors. Instead, we aim to bridge the sim-to-real gap of *raw sensor readings* (Figure 1).

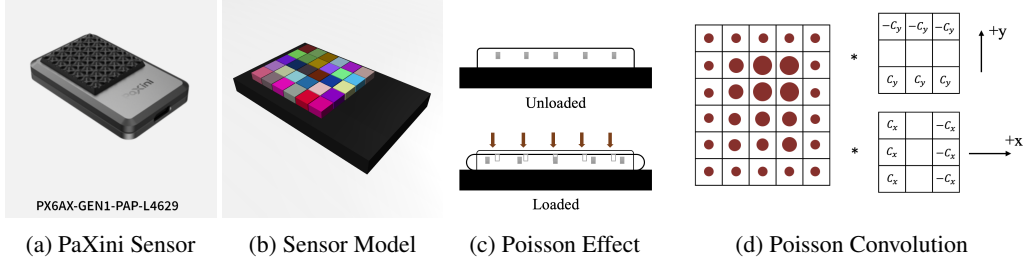


Figure 2: (a) Product PX6AX-GEN1-PAP-L4629 from PaXini Tech, figure from [19]. (b) MJCF visualization of the tactile sensor model in simulation. We use  $6 \times 5$  small cubes instead of a single cube. (c) Illustration of Poisson Effect on the tactile sensor in real world. Gray cubes represent the small magnets injected in the gel [12]. (d) Illustration of convolution technique to approximate Poisson Effect noise in real sensor readings. Red dots represent the normal force on each taxel, which is convolved with poisson-effect kernels in both directions.  $c_{x,y}$  are hyper-parameters that control the noise scale.

To achieve this, we first identify key aspects of the sim-to-real gap: non-uniform contact, contact Poisson Effect, and force scaling difference. We then develop simple yet efficient techniques to mitigate these gaps (Figure 3): surface with Gaussian bumps, Convolution for Poisson Effect and domain randomization for Force Scaling. In our experiment, we show that our approach GCS enables *zero-shot* sim-to-real transfer of RL policy on *blind* peg-in-hole insertion tasks, i.e., the robot does not have accurate peg poses. Results suggest that our GCS method outperforms all previous sim-to-real techniques and improves the success rate by **50%**. For some tasks, our RL policy succeeds on 9 out of 10 real-world trials.

## 2 Method

In our work, we use the dense, distributed, 3-axes, magnet-based tactile sensor: PX6AX-GEN1-PAP-L4629 (Figure 2a) from PaXini Tech [19]. To simulate tactile sensors, we extend Robosuite [20] which is built on the MuJoCo [21] physics engine. Similar to Sferazza et al. [22], we use the Touch Grid plugin provided by MuJoCo to get the aggregate contact forces in each taxel area. In addition, as shown in Figure 2b, instead of using a single cube as the contact pad of the sensor, we discretize the contact pad with small cubes of  $4.6mm \times 4.6mm \times 1mm$  ( $W \times H \times D$ ).

Figure 1 compares tactile readings in MuJoCo with readings from real world sensors under the same physical state. In the first row, the robot holds the cubic peg handle, which has no contact with external environment. In the second row, the robot holds the same handle, while the peg is in contact with the back rim of the square hole. There exists a significant sim-to-real gap in tactile sensor readings. Please refer to the appendix for an intuitive interpretation of the gap.

### 2.1 GCS: Bridge Sim-to-Real Gap

Prior work bridges the sim-to-real gap via binarization techniques [9, 23, 10]. However, this may lose information that is crucial in other tasks. In our work, we aim to bridge the sim-to-real gap of *raw tactile readings* of dense, distributed, 3-axes tactile forces, as it is the most general form of tactile input. To this end, we propose three simple yet effective techniques to address the above gaps.

**1. Surface with Gaussian Bumps.** We randomize the depth of the small cubes on the surface pad in simulation. Specifically speaking, we randomly select a 2-dimensional Gaussian distribution, i.e., mean  $(g_x, g_y) \in \mathcal{G} = [0, 5] \times [0, 6]$  and standard deviation  $(s_x, s_y)$ . Then, given the coordinate of each taxel  $(i, j) \in \mathcal{G}$ , we denote  $D((i, j), (g_x, g_y)) = \sqrt{(g_x - i)^2/s_x^2 + (g_y - j)^2/s_y^2}$ . Then, the depth  $h$  of taxel cube  $(i, j)$  is

$$h(i, j) = h_{\min} + \frac{D((i, j), (g_x, g_y))}{\max_{(i, j) \in \mathcal{G}} D((i, j), (g_x, g_y))} * (h_{\max} - h_{\min}).$$

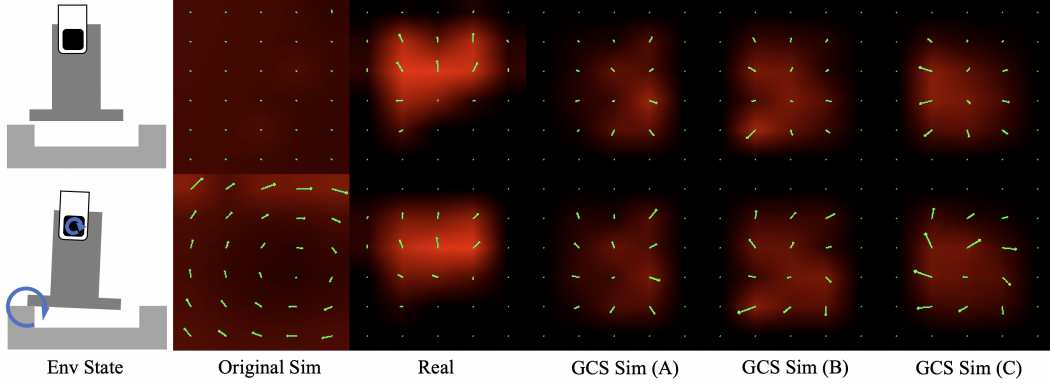


Figure 3: For right 3 figures, we show the *left* finger GCS tactile readings (A)(B)(C) under 3 random parameters sampled from Table 3, and compare them with the real-world tactile readings and the original raw simulation tactile readings under the same physical states as in Figure 1.

Here  $h_{\max}$  and  $h_{\min}$  denotes the maximal and minimal cube depth, which are chosen as hyper-parameters.

**2. Convolution for Poisson Effect.** We approximate the Poisson Effect noise via a simple approach, avoiding time-consuming methods like FEM simulation. Observing that the scale of the shear force is proportional to the normal force in its surroundings, we approximate the Poisson Effect shear displacement by convolving the normal forces tensor with poisson-effect kernels, illustrated in Figure 2d. The poisson-effect scale is determined by parameters  $c_x, c_y$  respectively for each direction.

**3. Domain Randomization of Force Scaling.** We randomize the force scaling factors in each episode during training. To be specific, we randomize the scaling for each axis of the all taxels, i.e.,  $\alpha_k, k \in [x, y, z]$ , and independently randomize the scale of each taxel  $\beta_{ij}, (i, j) \in \mathcal{G}$ . Namely, each taxel reading  $f_k(i, j) = \hat{f}_k(i, j) * \alpha_k * \beta_{ij}$ , where  $\hat{f}_k(i, j)$  is the raw force reading from the physics engine for taxel  $(i, j)$  in direction  $k$ .

Figure 3 illustrates the revised tactile readings with GCS, in the same physical states as Figure 1.

### 3 Experiment

**Task.** We use the following 6 *blind* insertion tasks in our experiment (Figure 6). The robot is *blind* so that it does not know the pose of the peg, while the peg is randomly placed between fingers at each initialization. Consequently, the robot is required to infer contact between the peg and the hole base from tactile sensing. Please refer to the appendix for task details.

- **RY-2mm:** Round peg with cylinder handle, and 2mm clearance round hole.
- **RU-2mm:** Round peg with cubic handle, and 2mm clearance round hole.
- **SQ-2mm:** Square peg with cubic handle, and 2mm clearance square hole.
- **SQ-1mm:** Square peg with cubic handle, and 1mm clearance square hole.
- **SX-2mm:** Square peg with cubic handle, and 2mm clearance groove along the x-axis. In addition, the hole pose is also blind to the robot.
- **SY-2mm:** Square peg with cubic handle, and 2mm clearance groove along the y-axis. In addition, the hole pose is also blind to the robot.

**Baselines.** We compare GCS with the following baselines, that are trained with asymmetric SAC. Implementation details are listed in the appendix.

1. **No Tactile (NT).** The policy  $\pi(p_t)$  is a 2-layer MLP that does not use tactile information.
2. **Total Force (TF).** Total contact force  $f_t$  on each surface pad is computed by summing all taxel readings. The force input  $f_t$  is a  $6k$ -dim vector, which is concatenated with  $p_t$ , i.e.,  $\pi(p_t, f_t)$ .
3. **Taxel Binarization (TB).** Similar as [23, 9], we binarize the taxel reading with a 0.1N threshold. The tactile input is a  $2k \times 6 \times 5$  tensor of  $[0, 1]$ . We encode with a similar CNN.

- 85 4. **Taxel Direction (TD)**. Inspired by [10], we binarize each taxel reading in each axis with 0.1N  
 86 threshold, together with the sign to represent the direction. Thus, the tactile input is a  $6k \times 6 \times 5$   
 87 tensor of value  $[-1, 0, 1]$ , encoded with a similar CNN.
- 88 5. **Direct Transfer (DT)**. We use the tactile reading from the simulator (Figure 1) directly, which  
 89 is a  $6k \times 6 \times 5$  tensor, and is encoded with a similar CNN encoder.
- 90 **Results**. For real-world evaluation, we test each method with 10 randomly initialized trials. In  
 91 simulation, we test with 50 trials. The result is shown in Table 1.

Table 1: Comparative evaluation on success rate of baselines and ours GCS in simulation and real world. NT: No Tactile; TF: Total Force; TB: Taxel Binarization; TD: Taxel Direction; DT: Direct Transfer.

Method	RY-2mm	RU-2mm	SQ-2mm	SQ-1mm	SX-2mm	SY-2mm	Avg
NT (Sim)	0.94	0.82	0.6	0.56	0.42	0.2	0.59
NT (Real)	0.4	0.4	0.0	0.1	0.0	0.0	0.15
TF (Sim)	1.0	0.98	0.8	0.0	1.0	0.74	0.75
TF (Real)	0.4	0.3	0.3	0.0	0.4	0.1	0.25
TB (Sim)	1.0	0.82	0.62	0.36	0.84	0.1	0.62
TB (Real)	0.5	0.3	0.4	0.1	0.5	0.2	0.33
TD (Sim)	1.0	0.86	0.9	0.88	0.88	0.80	0.88
TD (Real)	0.1	0.4	0.0	0.2	0.2	0.2	0.18
DT (Sim)	1.0	0.96	0.84	0.94	0.96	0.84	0.92
DT (Real)	0.6	0.2	0.1	0.3	0.4	0.1	0.28
GCS (Sim)	1.0	0.9	0.84	0.98	0.98	0.76	0.91
GCS (Real)	<b>0.9</b>	<b>0.8</b>	<b>0.9</b>	<b>0.6</b>	<b>0.9</b>	<b>0.7</b>	<b>0.8</b>

92 We find that baselines cannot produce good policies that transfer well to the real world. For NT, it  
 93 has only  $<60\%$  success rate on square peg tasks even in simulation and in real world, it has  $<10\%$   
 94 success rate. For TF, it has a higher success rate in simulation than NT. However, as discussed in  
 95 Section 2, the real-world reading is quite noisy. We observe that the total force on the pad can even  
 96 point upward when there is no external contact. Consequently, the policy cannot perform well in  
 97 real world.

98 Binarization techniques (TB, TD) are useful for in-hand orientation tasks [9, 23] translation [10],  
 99 we find that they are not useful for insertion tasks. The real-world shear force noise makes the  
 100 feature in these methods non-transferrable to the real-world rollouts. In terms of DT, owing to large  
 101 sim-to-real gap (Figure 1row), the real-world success rate also is low due to distribution shift.

102 We observe that GCS outperforms all baselines by a 50% margin on average. In many tasks such as  
 103 RY-2mm, SQ-2mm, and SX-2mm, it has a real-world success rate of 90% with barely any perfor-  
 104 mance drop than in the simulation. The largest sim-to-real gap occurs in the SQ-1mm case, which  
 105 is the hardest task owing to the small clearance. On SQ-1mm, one needs fine-grained and compliant  
 106 control. Small action error, potentially from sim-to-real tactile gap, will cause the peg to miss the  
 107 hole in the next step.

## 108 4 Conclusion

109 Dense, distributed, 3-axes, magnet-based tactile sensor (i.e., u-skin) balances well between dura-  
 110 bility, tactile density, and compactness. In our work, we try to bridge the sim-to-real gap of these  
 111 sensors. Here, we identify three aspects of the sim-to-real gap: non-uniform contact, poisson effect,  
 112 and scale difference, and propose a novel GCS method to mitigate these gaps. We evaluate our  
 113 approach on *blind* insertion tasks and show that GCS improves the real-world success rate by 50%  
 114 over baselines. Our work creates new possibilities to future sim-to-real learning of visual-tactile  
 115 manipulation skills with magnet-based sensors.



## References

- [1] W. Yuan, S. Dong, and E. H. Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12):2762, 2017.
- [2] I. H. Taylor, S. Dong, and A. Rodriguez. Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10781–10787, 2022. doi:10.1109/ICRA46639.2022.9811832.
- [3] R. Kolamuri, Z. Si, Y. Zhang, A. Agarwal, and W. Yuan. Improving grasp stability with rotation measurement from tactile sensing. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6809–6816. IEEE, 2021.
- [4] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine. More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and Automation Letters*, 3(4):3300–3307, 2018.
- [5] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson. Cable manipulation with a tactile-reactive gripper. *The International Journal of Robotics Research*, 40(12-14):1385–1401, 2021.
- [6] A. Wilson, H. Jiang, W. Lian, and W. Yuan. Cable routing and assembly using tactile-driven motion primitives. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10408–10414. IEEE, 2023.
- [7] J. Zhao, M. Bauzá, and E. H. Adelson. Fingerslam: Closed-loop unknown object localization and reconstruction from visuo-tactile feedback. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8033–8039. IEEE, 2023.
- [8] S. Suresh, H. Qi, T. Wu, T. Fan, L. Pineda, M. Lambeta, J. Malik, M. Kalakrishnan, R. Calandra, M. Kaess, et al. Neuralfeels with neural fields: Visuotactile perception for in-hand manipulation. *Science Robotics*, 9(96):ead10628, 2024.
- [9] H. Qi, B. Yi, S. Suresh, M. Lambeta, Y. Ma, R. Calandra, and J. Malik. General in-hand object rotation with vision and touch. In *Conference on Robot Learning*, pages 2549–2564. PMLR, 2023.
- [10] J. Yin, H. Qi, J. Malik, J. Pikul, M. Yim, and T. Hellebrekers. Learning in-hand translation using tactile skin with shear and normal force sensing. *arXiv preprint arXiv:2407.07885*, 2024.
- [11] B. Huang, Y. Wang, X. Yang, Y. Luo, and Y. Li. 3d-vitac: Learning fine-grained manipulation with visuo-tactile sensing. *arXiv preprint arXiv:2410.24091*, 2024.
- [12] T. P. Tomo, W. K. Wong, A. Schmitz, H. Kristanto, A. Sarazin, L. Jamone, S. Somlor, and S. Sugano. A modular, distributed, soft, 3-axis sensor system for robot hands. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 454–460. IEEE, 2016.
- [13] T. P. Tomo, M. Regoli, A. Schmitz, L. Natale, H. Kristanto, S. Somlor, L. Jamone, G. Metta, and S. Sugano. A new silicone structure for uskin—a soft, distributed, digital 3-axis skin sensor and its integration on the humanoid robot icub. *IEEE Robotics and Automation Letters*, 3(3): 2584–2591, 2018.
- [14] R. Bhirangi, V. Pattabiraman, E. Erciyes, Y. Cao, T. Hellebrekers, and L. Pinto. Anyskin: Plug-and-play skin sensing for robotic touch. *ICRA*, 2025.
- [15] J. Lenz, I. Pfenning, T. Gruner, D. Palenicek, T. Schneider, and J. Peters. Exploring the role of vision and touch in reinforcement learning for dexterous insertion tasks.

- [16] N. Funk, E. Helmut, G. Chalvatzaki, R. Calandra, and J. Peters. Evetac: An event-based optical tactile sensor for robotic manipulation. *IEEE Transactions on Robotics*, 2024.
- [17] I. Akinola, J. Xu, J. Carius, D. Fox, and Y. Narang. Tacsl: A library for visuotactile sensor simulation and learning. *IEEE Transactions on Robotics*, 2025.
- [18] Px-6ax gen3 hand. <https://paxini.com/dex/gen3>, . Accessed: 2025-08-25.
- [19] Px-6ax gen1. <https://paxini.com/ax/gen1>, . Accessed: 2025-04-23.
- [20] Y. Zhu, J. Wong, A. Mandlekar, R. Martín-Martín, A. Joshi, S. Nasiriany, and Y. Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.
- [21] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.
- [22] C. Sferrazza, Y. Seo, H. Liu, Y. Lee, and P. Abbeel. The power of the senses: Generalizable manipulation from vision and touch through masked multimodal learning. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9698–9705. IEEE, 2024.
- [23] Z.-H. Yin, B. Huang, Y. Qin, Q. Chen, and X. Wang. Rotating without seeing: Towards in-hand dexterity through touch. *arXiv preprint arXiv:2303.10880*, 2023.
- [24] M. Li, L. Zhang, Q. Huang, T. Li, and Y. Jiang. Modeling, simulation, and application of spatio-temporal characteristics detection in incipient slip. *arXiv preprint arXiv:2502.17335*, 2025.
- [25] N. Sunil, S. Wang, Y. She, E. Adelson, and A. R. Garcia. Visuotactile affordances for cloth manipulation with local control. In *Conference on Robot Learning*, pages 1596–1606. PMLR, 2023.
- [26] S. Dong, D. K. Jha, D. Romeres, S. Kim, D. Nikovski, and A. Rodriguez. Tactile-rl for insertion: Generalization to objects of unknown geometry. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6437–6443. IEEE, 2021.
- [27] S. Kim and A. Rodriguez. Active extrinsic contact sensing: Application to general peg-in-hole insertion. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 10241–10247. IEEE, 2022.
- [28] K. Yu, Y. Han, Q. Wang, V. Saxena, D. Xu, and Y. Zhao. Mimictouch: Leveraging multi-modal human tactile demonstrations for contact-rich manipulation. *arXiv preprint arXiv:2310.16917*, 2023.
- [29] S. Kim, A. Bronars, P. Patre, and A. Rodriguez. Texterity–tactile extrinsic dexterity: Simultaneous tactile estimation and control for extrinsic dexterity. *arXiv preprint arXiv:2403.00049*, 2024.
- [30] 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. IEEE, 2024.
- [31] R. Bhirangi, T. Hellebrekers, C. Majidi, and A. Gupta. Reskin: versatile, replaceable, lasting tactile skins. In *5th Annual Conference on Robot Learning*.
- [32] V. Pattabiraman, Z. Huang, D. Panozzo, D. Zorin, L. Pinto, and R. Bhirangi. eflesh: Highly customizable magnetic touch sensing using cut-cell microstructures. *arXiv preprint arXiv:2506.09994*, 2025.

- [33] Y. Yuan, H. Che, Y. Qin, B. Huang, Z.-H. Yin, K.-W. Lee, Y. Wu, S.-C. Lim, and X. Wang. Robot synesthesia: In-hand manipulation with visuotactile sensing. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6558–6565. IEEE, 2024.
- [34] B. Huang, J. Xu, I. Akinola, W. Yang, B. Sundaralingam, R. O’Flaherty, D. Fox, X. Wang, A. Mousavian, Y.-W. Chao, and Y. Li. Vt-refine: Learning bimanual assembly with visuotactile feedback via simulation fine-tuning.
- [35] 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. *arXiv preprint arXiv:2409.17549*, 2024.
- [36] I. Guzey, Y. Dai, B. Evans, S. Chintala, and L. Pinto. See to touch: Learning tactile dexterity through visual incentives. *arXiv preprint arXiv:2309.12300*, 2023.
- [37] I. Guzey, B. Evans, S. Chintala, and L. Pinto. Dexterity from touch: Self-supervised pre-training of tactile representations with robotic play. *arXiv preprint arXiv:2303.12076*, 2023.
- [38] A. Sharma, C. Higuera, C. K. Bodduluri, Z. Liu, T. Fan, T. Hellebrekers, M. Lambeta, B. Boots, M. Kaess, T. Wu, et al. Self-supervised perception for tactile skin covered dexterous hands. *arXiv preprint arXiv:2505.11420*, 2025.
- [39] F. Yang, C. Feng, Z. Chen, H. Park, D. Wang, Y. Dou, Z. Zeng, X. Chen, R. Gangopadhyay, A. Owens, et al. Binding touch to everything: Learning unified multimodal tactile representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26340–26353, 2024.
- [40] T. Chen, M. Tippur, S. Wu, V. Kumar, E. Adelson, and P. Agrawal. Visual dexterity: In-hand reorientation of novel and complex object shapes. *Science Robotics*, 8(84):eadc9244, 2023. doi:10.1126/scirobotics.adc9244. URL <https://www.science.org/doi/abs/10.1126/scirobotics.adc9244>.
- [41] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State. Isaac gym: High performance gpu-based physics simulation for robot learning, 2021.
- [42] Z. Si and W. Yuan. Taxim: An example-based simulation model for gelsight tactile sensors. *IEEE Robotics and Automation Letters*, 7(2):2361–2368, 2022.
- [43] S. Wang, M. Lambeta, P.-W. Chou, and R. Calandra. Tacto: A fast, flexible, and open-source simulator for high-resolution vision-based tactile sensors. *IEEE Robotics and Automation Letters*, 7(2):3930–3937, 2022.
- [44] Z. Si, G. Zhang, Q. Ben, B. Romero, Z. Xian, C. Liu, and C. Gan. Diff tactile: A physics-based differentiable tactile simulator for contact-rich robotic manipulation. *arXiv preprint arXiv:2403.08716*, 2024.
- [45] J. Xu, S. Kim, T. Chen, A. R. Garcia, P. Agrawal, W. Matusik, and S. Sueda. Efficient tactile simulation with differentiability for robotic manipulation. In *Conference on Robot Learning*, pages 1488–1498. PMLR, 2023.
- [46] W. Du, W. Xu, J. Ren, Z. Yu, and C. Lu. Tacipc: Intersection-and inversion-free fem-based elastomer simulation for optical tactile sensors. *IEEE Robotics and Automation Letters*, 9(3):2559–2566, 2024.
- [47] W. Chen, J. Xu, F. Xiang, X. Yuan, H. Su, and R. Chen. General-purpose sim2real protocol for learning contact-rich manipulation with marker-based visuotactile sensors. *IEEE Transactions on Robotics*, 40:1509–1526, 2024.

- 247 [48] Y. Li, W. Du, C. Yu, P. Li, Z. Zhao, T. Liu, C. Jiang, Y. Zhu, and S. Huang. Taccel: Scal-  
248 ing up vision-based tactile robotics via high-performance gpu simulation. *arXiv preprint*  
249 *arXiv:2504.12908*, 2025.
- 250 [49] M. Li, Z. Ferguson, T. Schneider, T. R. Langlois, D. Zorin, D. Panozzo, C. Jiang, and D. M.  
251 Kaufman. Incremental potential contact: intersection-and inversion-free, large-deformation  
252 dynamics. *ACM Trans. Graph.*, 39(4):49, 2020.
- 253 [50] B. Tang, M. A. Lin, I. Akinola, A. Handa, G. S. Sukhatme, F. Ramos, D. Fox, and Y. Narang.  
254 Industreal: Transferring contact-rich assembly tasks from simulation to reality. *arXiv preprint*  
255 *arXiv:2305.17110*, 2023.
- 256 [51] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu. Viola: Imitation learning for vision-based manipulation  
257 with object proposal priors. *arXiv preprint arXiv:2210.11339*, 2022. doi:10.48550/arXiv.2210.  
258 11339.
- 259 [52] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy  
260 deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International*  
261 *Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*,  
262 pages 1861–1870, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.
- 263 [53] Stablebaseline3 documentation. [https://stable-baselines3.readthedocs.io/en/](https://stable-baselines3.readthedocs.io/en/master/)  
264 [master/](https://stable-baselines3.readthedocs.io/en/master/). Accessed: 2025-04-23.

## A Related Works

**Robot Manipulation with Tactile Sensors.** Tactile sensors are crucial for many manipulation tasks. They provide information of contact between robots and objects, and contact between objects in hand and its surroundings. Moreover, such information cannot be easily revealed from visual information alone. Many prior works used vision-based optical tactile sensors (e.g., Gelsight [1, 2]) in grasping [3, 4], slip detection [24], cable manipulation [5, 6], cloth manipulation [25], visual-tactile slam [7, 8], contact-rich insertion [26, 27, 28], in hand manipulation [9] and in exploiting external contact [29, 30]. Recently, works have also used piezoelectric and magnet-based sensors [14, 31, 32] for various manipulation tasks, including visual-tactile in-hand reorientation and translation [23, 33, 10]. Most importantly, a piezoelectric sensor, i.e., 3D-ViTac [11, 34] and a magnet-based sensor, e.g., u-skin [12, 13] have been introduced to robot manipulation. These two sensors can provide dense, distributed tactile signals. Works have used these high-dimensional tactile information for complex manipulation tasks via imitation learning on tele-operated demonstrations [11, 35, 36, 37, 38]. Unlike these prior work, we focus on bridging the sim-to-real gap of dense, distributed, 3-axes, magnet-based sensors.

**Tactile Sensor Sim-to-Real.** There is only limited data involving real-world tactile readings [39], e.s.p., data collected during robot manipulation [29, 11]. In contrast, simulation data are relatively cheap to acquire and work has shown learning dexterous skills with only simulation data [40, 9, 23]. However, as the mechanisms of the rigid-body physics engines [21, 41] are different from the real world physics, bridging the large sim-to-real gap for manipulation with tactile sensors remains a challenge. For optical tactile sensors, prior work developed example-based simulators [42, 43], differentiable simulators [44], rigid-body simulators coupled with penalty-based models [17, 45] and FEM-based simulators [46, 47, 48]. Notably, Chen et al. [47] achieved zero-shot sim-to-real transfer on contact-rich tasks [47] with IPC-based [49] simulation. In addition, works have also shown success on dexterous in-hand orientation [23, 9] and translation [10], with binary tactile signals from magnet-based or piezoelectric sensors. However, binarization loses a lot of information from dense, distributed, 3-axes sensors. For example, in peg-in-hole insertion tasks, binarization does not provide information of contact between the peg and the hole base. In our work, we use the raw tactile reading directly as input to the RL policy and we develop a different technique to bridge the sim-to-real gap of raw tactile reading for dense, distributed, 3-axes, magnet-based sensors.

## B Additional Results

### B.1 Ablation Study

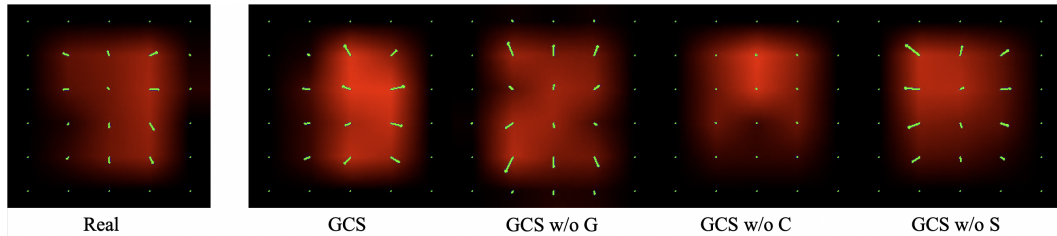


Figure 4: Visualization of ablation’s left tactile readings in the no contact state. Noticing that readings in GCS and ablations are randomized in training.

Here, we demonstrate the necessity of all  $G$ ,  $C$ ,  $S$  technique to bridge the sim-to-real gap.

Table 2 shows the real-world success rate of the ablation. In simulation, the success rate is  $>80\%$  for all ablations. Clearly, results show that all three  $G, C, S$  techniques are necessary. This is especially the case in SX-2mm and SY-2mm tasks, where it only uses tactile reading as input. Consequently, distribution mismatch between simulated and real-world tactile readings will result in the failure of the contact-rich insertion task. Figure 4 visualizes the tactile reading in simulation of different ablations. With all  $G, C, S$  techniques, the reading looks more similar to real world.

Table 2: Ablation study of GCS on SX-2mm, SY-2mm, and SQ-2mm tasks. In each ablation, we remove one technique in Section 2.1.

	GCS	GCS w/o G	GCS w/o S	GCS w/o C
SX-2mm	0.9	0.4	0.3	0.3
SY-2mm	0.7	0.0	0.6	0.2
SQ-2mm	0.9	0.9	0.8	0.6

## 304 B.2 Policy Interpretation

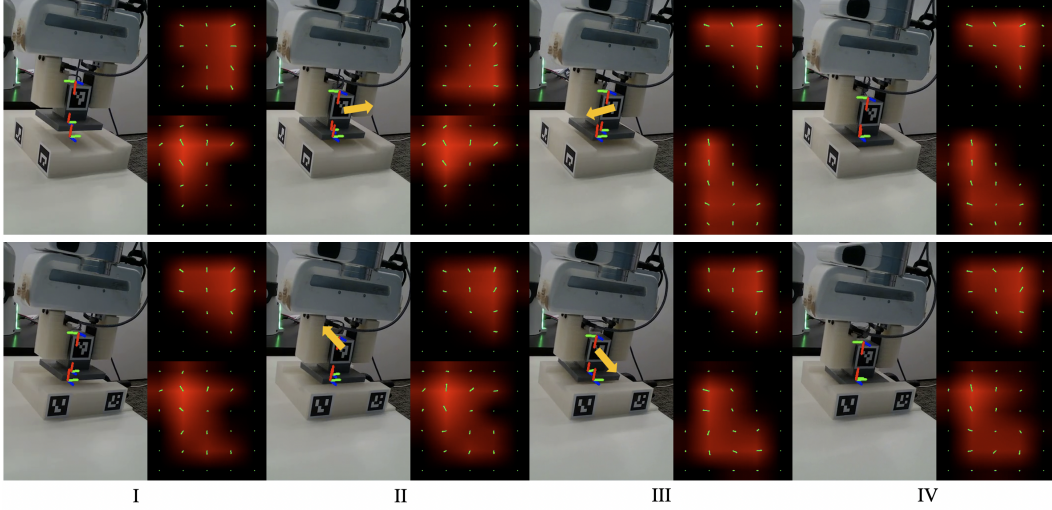


Figure 5: Examples of policy rollout in SX-2mm (first row) and SY-2mm (second row). Stage I corresponds to the initial state, and Stage IV corresponds to the success state. Stage II and Stage III corresponds to the critical contact states that the policy infers the relative position from (Section B.2).

305 To better understand how the policy trained in simulation transfers to real world, we visualize tra-  
 306 jectories in SX-2mm and SY-2mm tasks, in Figure 5. In SY-2mm, it detects the relative position  
 307 between the peg and the hole by the change of the shear forces. In stage II, the peg contacted with  
 308 the front rim and thus shear force markers on the left finger changed counterclockwise and clock-  
 309 wise on the right finger sensor. Consequently, the policy commanded the hand to move backward.  
 310 However, it overshoot and contacted the back rim (stage III). Again, it detected the opposite change  
 311 on both sensors and thus moved forward. This time, it managed to insert the peg in the groove.

312 For SX-2mm, it detects the relative position by the change of distribution of normal force on the  
 313 sensor pads. In stage II, it contacted the left rim and the normal force shifted downward on the left  
 314 sensor, owing to the external torque. The policy then commanded it to move right. In stage III, it  
 315 contacted with the right rim and the normal force on the right pad shifted downward this time. Then  
 316 it moved left in fine-grained steps until the peg was inserted into the groove.

## 317 C Experiment Details

318 **Sim-to-Real Gap.** We identify that the gap comes from three main aspects:

- 319 1. **Non-uniform Contact.** The actual surface gel is uneven and the injected small magnets [12]  
 320 are not completely symmetric in terms of magnetic field. Therefore, we observe non-uniform  
 321 activation of the dense distributed Hall Effect sensors, i.e., non-uniform normal force readings,  
 322 despite that the contact surface is flat.



- 323 2. **Contact Poisson Effect.** There exists a significant non-zero shear force, even when the peg is  
 324 not in contact with the environment. We find that this results from the Poisson Effect of the  
 325 contact gel, which is illustrated in Figure 2c.
- 326 3. **Different Force Scale.** We find that the absolute scale of real-world sensor force differs from  
 327 the scale in simulation. This comes from the difference between the physics engine and real-  
 328 world physics.
- 329 **Randomization Ranges** Table 3 summarizes the randomization range of all factors in our pipeline.

Table 3: Domain Randomization (DR) distribution of key parameters in our GCS method.

Parameters	Axes Force $\alpha_k$	Taxel Force $\beta_{ij}$	Convolution $c_{x,y}$	Deviation $s_{x,y}$
DR Range	$\mathcal{U}(0.5, 1.5)$	$\mathcal{N}(1.0, 0.25)$ , 20% Dropout	$\mathcal{U}(0.1, 0.3)$	$\mathcal{U}(1.0, 3.0)$

330

331 **Task.** We use Operation Space Controller (OSC) at 20Hz. For RY-2mm, RU-2mm, SQ-2mm, SQ-  
 332 1mm, the policy network outputs 3-dim position delta (XYZ) in  $[-1, 1]$ , scaled with 1cm, except for  
 333 SQ-1mm, which is scaled with 5mm. Here, we freeze the rotation in OSC to simplify the tasks. For  
 334 SX-2mm, we further freeze X-axis and for SY-2mm, we freeze Y-axis.

335 At initialization of each episode in simulation, we randomize the parameters in Table 3. Similar to  
 336 [50], we move the gripper to the peg grasp pose with small randomization before the policy rollout.  
 337 In real world, we first command a randomized initial fingertip pose. Then we place the peg in the  
 338 gripper with small position randomization.

339 For real-world evaluation, we will consider success insertion if the bottom of the peg is in the hole.  
 340 Each trial is truncated at 120 steps for SX-2mm and SY-2mm, and 200 steps for RY-2mm, RU-2mm,  
 341 SQ-2mm, and SQ-1mm.

342 **Reward.** The RL reward for all tasks is defined with the following 4 terms.

- 343 - *Reaching Reward.* We compute the average distance  $d$  of keypoints of the peg to correspondent  
 344 points on base. The reaching reward is  $r_{\text{reach}} = \frac{1}{10*d+\epsilon}$ .
- 345 - *Engagement Reward.* Given the engagement distance  $h$  (the distance between the bottom of peg  
 346 to the base bottom), if the peg is engaged with the hole, then we have  $r_{\text{engage}} = \frac{1}{10*h+\epsilon}$ , otherwise  
 347  $r_{\text{engage}} = 0$ .
- 348 - *Success Bonus.* The success bonus  $r_{\text{succ}}$  if  $h < h_{\text{th}}$ , where  $h_{\text{th}}$  is a threshold value.
- 349 - *Action Penalty.* We penalize the action scale with  $r_{\text{action}} = -\|a\|^2$ .

350 The total reward is the weighted sum of these 4 terms. The weights are selected as hyper-paramters.

351 **Hardware.** We custom 3d-printed fingers to attach sensors to the Franka Research 3 Gripper (Fig-  
 352 ure 6). We place Realsense L515 in front of the scene to capture object poses, which are labelled  
 353 with Aruco markers. All poses are computed relative to the robot base frame with camera calibra-  
 354 tion. Noticing that peg poses are not available to the robot and base poses in SX-2mm and SY-2mm  
 355 are also not available. Here, we put markers for tracking and task initialization. We use Deoxys [51]  
 356 to run the RL policy with Operation Space Controller (OSC) on the real robot.

357 **RL Policy.** For all policies, the input includes relative pose from robot fingertip to the hole base  $p_t$   
 358 (zeros for SX-2mm, SY-2mm tasks). For GCS and baselines (Section ??) with tactile inputs, we use  
 359 a history of  $k$  tactile readings, where  $k$  is a hyperparameter for each method. In GCS, the tactile  
 360 input  $o_t$  is a  $6k \times 6 \times 5$  ( $C \times H \times W$ ) tensor. The policy network first encodes the tactile reading with  
 361 a CNN to a 128-dim latent vector  $t_t = T(o_t)$  and encodes  $h_t = E(p_t)$  to a 128-dim vector. Then,  
 362 we feed them into a 2-layer MLP action head  $a_t = \pi(E(p_t), T(o_t))$ . Here, we use asymmetric  
 363 SAC [52] to train the RL policy in Robosuite [20], implemented based on StableBaseline3 [53]. The  
 364 critic network takes ground truth pose of objects and fingertip poses as input, but without tactile  
 365 readings. For all tasks, the reward is the sum of reaching reward, engagement reward, and success  
 366 bonus, which is similar to [50]. Please refer to the appendix for RL training details.

367 **Training.** We select the optimal  $k$  in  $\{1, 5, 10, 20\}$  for all baselines. We choose  $k=1$  for TF,  $k=10$   
368 for TB, TD and DT. For GCS, we choose  $k=20$  for all tasks except SY-2mm, which we find  $k=10$   
369 works better. In Section B.1, GCS ablations use the same  $k$  as GCS.

370 The tactile tensor encoder  $T$  in TB, TD, DT, and GCS is a 4-layer CNN with BatchNorm and ReLU  
371 activation and the encoder for  $p_t$  is a single-layer MLP. All MLP policy heads are 256-dim, 2-layer  
372 MLPs. For the critic in SAC, it is a 2-layer MLP that takes relative pose from peg to base and  
373 end-effector to base as inputs. The implementation is based on StableBaseline3 [53].

## D Additional Figures

**Task Illustration.** Figure 6 shows the illustration examples of our experiments.

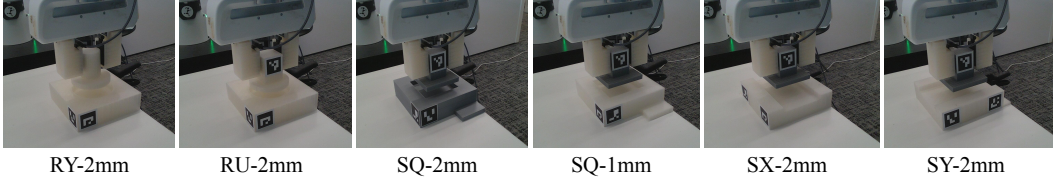


Figure 6: Illustration of all 6 *blind* insertion tasks in our experiment.

**Sensor Finger Mount.** Figure 7a and Figure 7b show the 3d-printed custom finger to mount the tactile sensor and its CAD model. We attach the sensor with double-sided tape on the fingers.

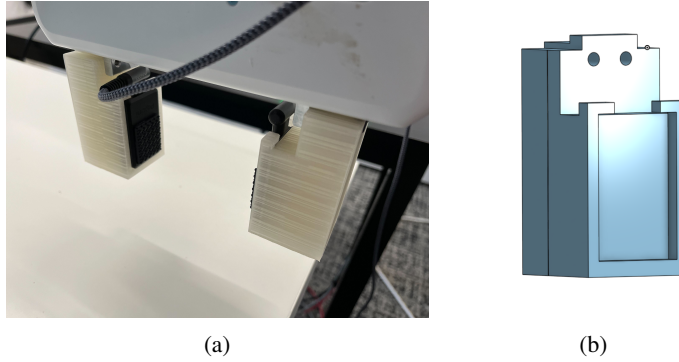


Figure 7: (a) The finger mount to attach the sensor on the Franka gripper. (b) CAD model for the custom finger.

**IsaacGym Tactile Reading.** Figure 8 shows the tactile rendering in Isaac-Gym [41] holding the square peg, with no contact with the external environment. The sensor model is similar as Figure 2b. Clearly, the tactile reading is noisy and unstructured. Therefore, we choose MuJoCo as our base simulator.

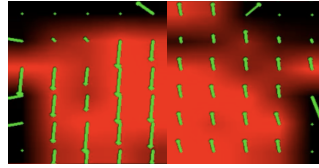


Figure 8: Tactile rendering in Isaac-Gym [41] with square peg at no contact state.

**Tactile Reading in RY-2mm.** RY-2mm use a round peg with cylindric handle. Consequently, the contact surface is quite narrow between. Thus, RY-2mm is a relatively easy task as certain pose information can be revealed from the tactile reading. Figure 9 compares GCS readings with real-world readings.

**Policy Interpretation on SQ-1mm.** With 1mm clearance, SQ-1mm is quite difficult and it requires soft, compliant motion. Figure 10 visualizes a policy rollout in SQ-1mm. In Stage II, the peg contacted with the back rim, which the policy managed to infer from small shear force change. Thus, it lifted the peg and moved forward. However, it overshoot and resulted in contact with the front rim (Stage III). This time, the policy lifted the peg and halt for a while instead of moving backward directly (Stage IV). Then it started to push the peg down, but it collided with the right rim and resulted in change in the distribution of normal force (Stage V). Then it moved left and downward. Here, it used very small actions and the peg was inserted successfully.

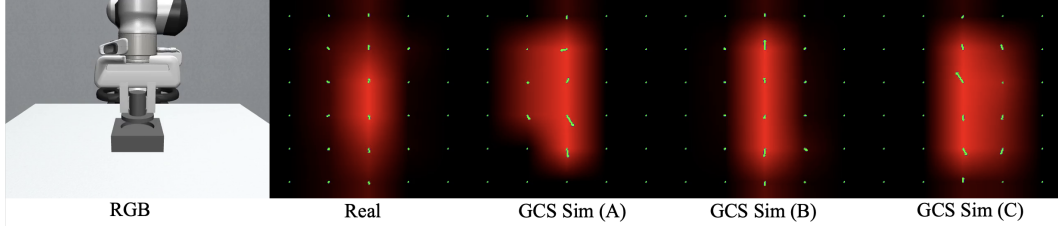


Figure 9: Tactile reading of RY-2mm task. The peg does not have contact with the base.

394 **Videos.** We have included example videos of real-world policy rollouts in the appendix.

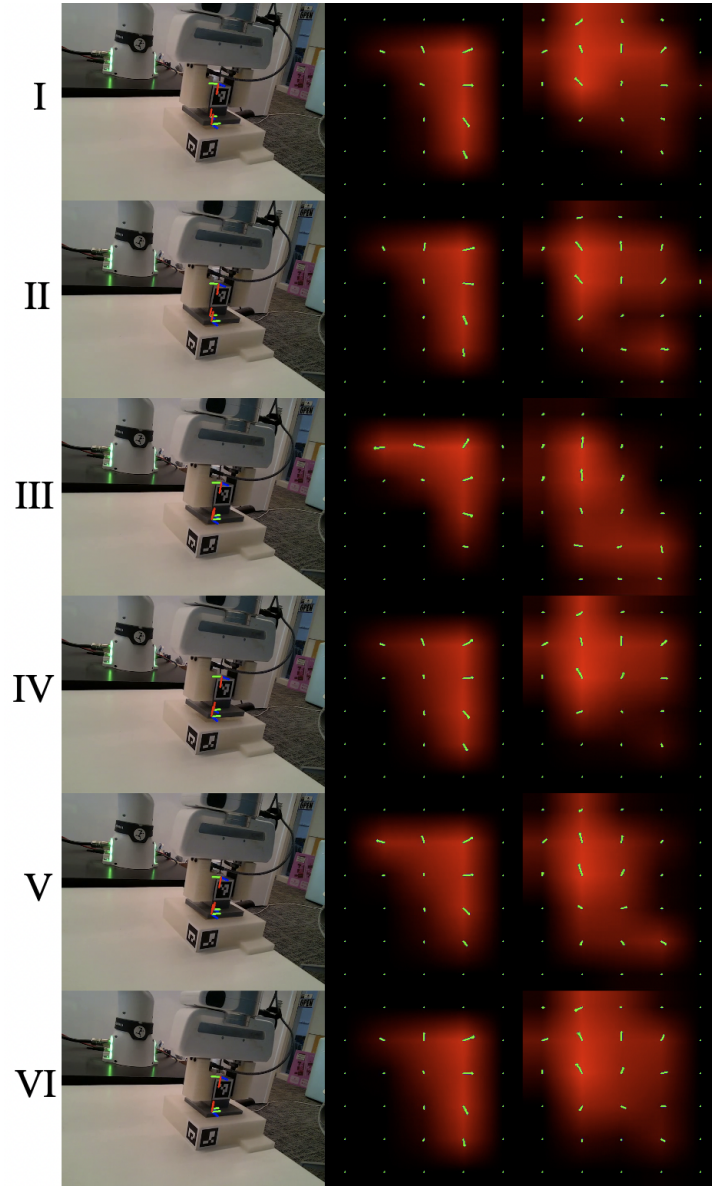


Figure 10: Policy rollout on SQ-1mm task.