

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

Beining Han¹, Abhishek Joshi¹, Jia Deng¹

Abstract—Tactile sensing is an important sensing modality for robot manipulation. Among different types of tactile sensors, magnet-based sensors, like u-skin, balance well between tactile density, high-durability, and compactness. However, the large sim-to-real gap of tactile sensors prevents robots from acquiring useful tactile-based manipulation skills from simulation data, a recipe that has been successful for achieving complex and sophisticated control policies. Prior work has used binarization techniques to bridge the sim-to-real gap for dexterous in-hand manipulation with magnet-based sensors. However, binarization inherently loses much information that is useful in many other tasks, e.g., insertion. In our work, we propose GCS, a novel sim-to-real technique to learn contact-rich insertion skills with dense, distributed, 3-axes tactile readings from magnet-based tactile sensors. We evaluated our approach on *blind* insertion tasks and show successful zero-shot sim-to-real transfer of RL policies with raw tactile readings as input. Code is released at: <https://github.com/princeton-v1/Tactile-GCS>.

I. INTRODUCTION

The sense of touch is an important sensing modality for humans. This sense enables humans to perform dexterous tasks such as using tools or detecting subtle changes in contact that can be critical for manipulating complex objects. Several efforts [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17] have aimed to replicate such tactile capabilities in robots. In particular, magnet-based tactile sensors such as u-skin [12], [13] provide dense tactile information, capturing both normal and shear forces. They are also durable and compact, enabling integration into dexterous hands [18].

Furthermore, training reinforcement learning policies in simulation has been an effective technique for learning robot skills [19], [20], [9], [21], [22], [23], [24]. The caveat to using simulation is the introduction of the sim-to-real gap. This gap is especially exacerbated for contact-rich tasks involving tactile sensors, making it difficult to transfer policies zero-shot to the real world. Although prior work has explored methods to bridge the sim-to-real gap for magnet-based tactile sensors, such as binarization [9], [10], these approaches discard dense tactile information and are not suitable for tasks such as blind insertion (Section IV).

In our work, we propose **GCS**, a novel method to bridge the gap between tactile readings from rigid-body simulation and the real world for magnet-based tactile sensors. GCS avoids using techniques such as binarization [10], [9] that are prone to information loss. Instead, we opt to bridge the sim-to-real gap directly for *dense, distributed tactile sensor readings* (Figure 1). Specifically, we show that GCS enables

the zero-shot transfer of RL policies from simulation for blind insertion tasks given raw tactile readings in the form of $6 \times 5 \times 3$ arrays as input.

We identify three key aspects of the sim-to-real gap for magnetic-based tactile sensors: non-uniform contact, contact Poisson Effect, and force scaling difference. For each aspect, we propose a simple yet effective technique to mitigate the gap (Section III-A). Our techniques involve introducing surfaces with Gaussian bumps, applying Convolutions for the Poisson Effect, and incorporating domain randomization for Force Scaling. Moreover, each technique is easy to implement and efficient to compute. **G** involves a simple randomized initialization of tactile sensor models, and **S**, **C** requires randomized data augmentation of the simulated tactile readings. Despite simplicity, we observe that the GCS readings are qualitatively more similar to their counterparts in the real world (Figure 3).

In our experiments, we show that GCS enables *zero-shot* sim-to-real transfer of RL policies on *blind* peg-in-hole insertion tasks, i.e., the robot does not have access to ground truth peg poses. As shown in Table II, the results suggest that GCS outperforms all previous sim-to-real techniques and improves the success rate by almost **50%**. On some tasks, our RL policy is successful in 9 out of 10 real-world trials, while baseline approaches failed completely. Furthermore, we find that each of the techniques that make up GCS is necessary for the final performance (Table III). To the best of our knowledge, GCS the first method that bridges the sim-to-real gap on blind insertion tasks of dense, 3-axes, magnet-based tactile sensors.

II. 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 [25], cable manipulation [5], [6], cloth manipulation [26], visual-tactile slam [7], [8], contact-rich insertion [27], [28], [29], in hand manipulation [9] and in exploiting external contact [30], [31]. Recently, works have also used piezoelectric and magnet-based sensors [14], [32], [33] for various manipulation tasks, including visual-tactile in-hand reorientation and translation [21], [34], [10]. Most importantly, a piezoelectric sensor, i.e., 3D-ViTac [11], [35] and a magnet-based sensor, e.g., u-skin [12], [13] have

¹ Princeton University bh7032@princeton.edu

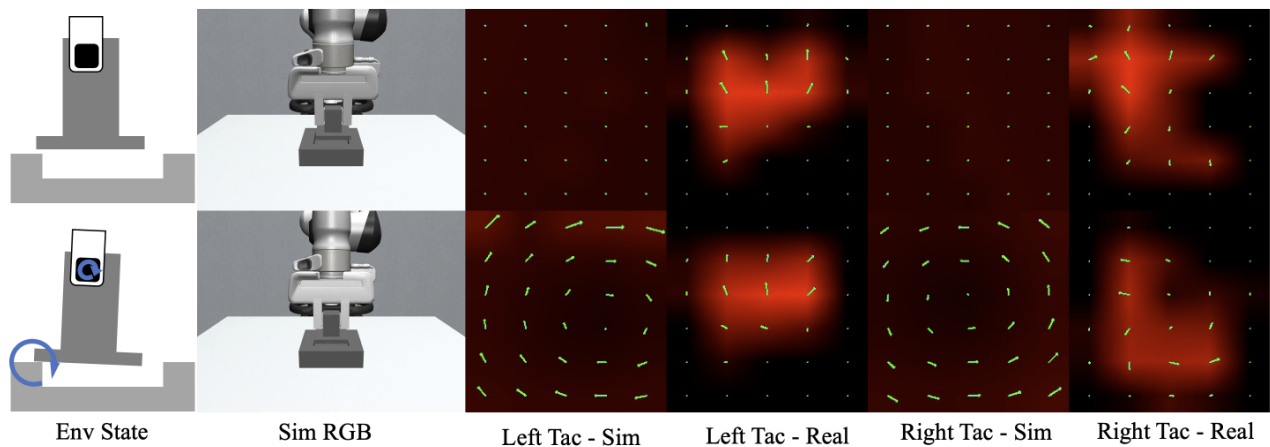


Fig. 1: Comparison of simulated 6×5 tactile readings in MuJoCo and in the real world under the same physical state. The first row is the state where the peg is not in contact with the base. The second row is the state where the peg is in contact with the back rim of the base. For the right four figures, 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.

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], [36], [37], [38], [39]. 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 [40], e.s.p., data collected during robot manipulation [30], [11]. In contrast, simulation data are relatively cheap to acquire and work has shown learning dexterous skills with only simulation data [23], [9], [21]. However, as the mechanisms of the rigid-body physics engines [41], [19] 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, [47] achieved zero-shot sim-to-real transfer on contact-rich insertion tasks [47] with IPC-based [49] simulation. However, these methods are not suitable for magnet-based sensors. In addition, works have also shown success on dexterous in-hand orientation [21], [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.

III. METHOD

In our work, we use the dense, distributed, 3-axes, magnet-based tactile sensor: PX6AX-GEN1-PAP-L4629 (Figure 2a) from PaXini Tech [50]. It is a commercial version of u-skin

sensor [12]. This sensor is more cost-effective compared to products from other manufacturers at roughly \$200 per pad. Moreover, these sensors are densely distributed, providing a 3-axes force reading of a 12×10 taxel grid on the $26.1mm \times 22.6mm$ surface area. This is significantly denser than most counterparts. Here, we down-sample the reading to 6×5 by 2×2 sum pooling, in order to improve the simulation speed in RL training.

To simulate tactile sensors, we extend Robosuite [51] which extends the MuJoCo [41] physics engine. MuJoCo features soft-contact and explicit dynamics integration. Compared to other common simulators, e.g., Isaac-Gym [19] or Bullet [52], this soft-contact modeling produces more consistent and stable force readings. Similar to [53], 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 $4.6mm \times 4.6mm \times 1mm$ ($W \times H \times D$). This produces dense contact force readings, while a single contact cube will result in a sparse contact force due to the underlying collision detection.

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 with no contact to its 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. The degree of redness reflects the amount of normal force on each taxel, while the green arrows represent the shear force direction and its scale. It is evident that there exists a significant sim-to-real gap in tactile sensor readings. We identify that the gap comes from three main aspects:

- 1) **Non-uniform Contact.** Due to a manufacturing defect, the actual surface gel is uneven and the injected small magnets [12] are not completely symmetric in terms of magnetic field. Therefore, we observe non-uniform activation of the dense, distributed Hall Effect sensors.

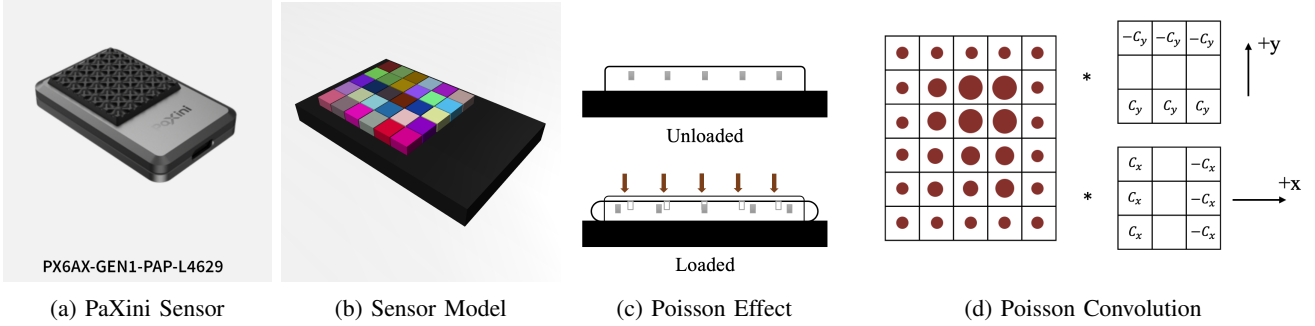


Fig. 2: (a) Product PX6AX-GEN1-PAP-L4629 from PaXini Tech, figure from [50]. (b) MJCF visualization of the tactile sensor model in simulation. We use 6×5 small cubes instead of a single cube. (c) Illustration of Poisson Effect on the tactile sensor in real world. Gray cubes represent the small magnets 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.

TABLE I: Domain Randomization (DR) distribution of key parameters in our GCS method.

Parameters	Axes Force α_k	Taxel Force β_{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	$\mathcal{U}(0.1, 0.3)$	$\mathcal{U}(1.0, 3.0)$

That is despite flat contact surfaces, we still observe non-uniform normal force readings. Moreover, such non-uniformity is inconsistent between different products. We note that left and right tactile readings may be different while their contact surfaces and gripper forces are symmetric.

- Contact Poisson Effect.** There exists a significant non-zero shear force even when the peg is not in contact with the environment. We find that this results from the Poisson Effect of the contact gel, which is illustrated in Figure 2c. This effect also leads to crosstalk among adjacent taxels [12], [13]. We find that the shear magnet displacement is in proportion with the scale of the loaded force in proximity.
- Different Force Scales.** The absolute scale of real-world sensor force differs from the scale in simulation. This comes from the difference between the physics engine and real-world physics. In MuJoCo, the contact force is determined by minimizing the next-step kinetic energy [41], which is subject to hyperparameters, like impulse regularization scaling and error reduction time constant. Different physical parameters will result in a different contact force scale in simulation. In addition, we also find that the real sensor reading is quite inaccurate.

A. GCS: Bridge Sim-to-Real Gap

Prior works bridge the sim-to-real gap via binarization techniques [9], [21], [10]. However, this technique loses information that is crucial for many tasks. For example, for insertion tasks (Section IV), binarization of taxel forces does not provide any information on the contact force between the peg and the environment. 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. To simulate the effect of non-uniform contact of the taxels, 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 $D((i, j), (g_x, g_y)) = \sqrt{(g_x - i)^2/s_x^2 + (g_y - j)^2/s_y^2}$. The depth $h(i, j)$ of taxel cube (i, j) is

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

Here, h_{\max} and h_{\min} denote the maximum and minimum cube depths, which are chosen as hyperparameters. Additionally, based on observations from real-world tactile readings, we reduce the depth at the boundary of the surface taxel grid. (Figure 1).

2. Convolution for Poisson Effect. High fidelity simulation of deformable objects require FEM-based approaches [49], which are computationally expensive. Here, we approximate the Poisson Effect noise via a simple approach. 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. To address the sim-to-real gap from inappropriate physical parameters and inaccurate sensor readings, we randomize the force scaling factors in each episode during training. Here, we randomize the scaling for each axis of 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 axis k .

Table I summarizes the randomization range of all factors in our pipeline. Figure 3 illustrates the revised tactile readings with GCS, in the same physical states as Figure 1. In Section IV, we show that the proposed method can bridge the sim-to-real gap for *blind* insertion tasks, allowing zero-shot transfer of an RL policy with raw tactile sensor readings as input.

IV. EXPERIMENT

In this section, we test GCS on transferring simulated tactile information to the real world for insertion tasks. Here, we aim to answer the following questions.

- 1) How does GCS perform compared to other sim-to-real techniques in previous work?
- 2) How does each technique in GCS contribute to real-world performance?

A. Experiment Setup

Task. We use the following 6 *blind* insertion tasks in our experiment (Figure 4).

- **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.

For each task, the robot does not know the pose of the peg, and the peg is randomly placed between its fingers during initialization. Consequently, the robot is required to infer contact between the peg and the hole base via tactile inputs. In our experiment, the robot hand is initialized such that the bottom of the peg is located within $[-1.5cm, 1.5cm] \times [1.5cm, 1.5cm] \times [0.5cm, 2cm]$ (XYZ) of the center of the hole. Please refer to the appendix for task details.

Hardware. We 3d-printed custom fingers to attach sensors to the Franka Research 3 Gripper (Figure 4). We place Realsense L515 in front of the scene to capture object poses, which are labeled with Aruco markers. All poses are computed relative to the robot base frame with camera calibration. Notice that in addition to peg poses, base poses in SX-2mm and SY-2mm are also not available to the robot. Here, we put markers for tracking and task initialization. We use Deoxys [54] to run the RL policy with an Operation Space Controller (OSC) at 20Hz on a real robot.

RL Policy. For all policies, the input includes the relative pose from the robot fingertip to the hole base p_t (zeros for SX-2mm, SY-2mm tasks). For GCS and baselines (Section IV-B) with tactile inputs, we use a history of k tactile

readings, where k is a hyperparameter for each method. In GCS, the tactile 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 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, we feed them into a 2-layer MLP action head $a_t = \pi(E(p_t), T(o_t))$. Here, we use asymmetric SAC [55] to train the RL policy in Robosuite [51], implemented based on StableBaseline3 [56]. The critic network takes ground truth pose of objects and fingertip poses as input without tactile readings. For all tasks, the reward is the sum of the reaching reward, engagement reward, and success bonus, which is similar to [57]. Please refer to the appendix for RL training details.

B. Comparative Evaluation

We compare GCS with the following baselines. Similar to GCS, each baseline is trained with asymmetric SAC (Section IV-A). 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 [21], [9], we binarize the taxel reading with a 0.1N threshold. The tactile input is a $2k \times 6 \times 5$ tensor of 0 and 1. We encode with a similar CNN.
- 4) **Taxel Direction (TD).** Inspired by [10], we binarize each taxel reading in each axis with 0.1N threshold, together with the sign to represent the direction. Thus, the tactile input is a $6k \times 6 \times 5$ tensor of value $[-1, 0, 1]$, encoded with a similar CNN.
- 5) **Direct Transfer (DT).** We use the tactile reading from the simulator (Figure 1) directly, which is a $6k \times 6 \times 5$ tensor, and is encoded with a similar CNN encoder.

For real-world evaluation, we test each method with 10 randomly initialized trials. In simulation, we test with 50 trials. The result is shown in Table II.

We find that baselines cannot produce good policies that transfer well to the real world. For NT, it has a $<60\%$ success rate on square peg tasks even in simulation. When transferred to the real world, it has a $<10\%$ success rate. We find that this comes from both pose tracking noise and OSC dynamics gap due to friction in physical motors. NT performs better on round peg tasks, as the round peg is rotation symmetric and thus is more compliant and tolerant to collision. In all, the low performance of NT suggests that *blind* insertion tasks require tactile information.

TF has a higher success rate in simulation than NT. However, as discussed in Section III, the real-world reading is quite noisy. We observe that the total force on the pad can even point upward when there is no external contact. Consequently, the policy cannot perform well in real world.

Though binarization techniques like TB are useful for in-hand orientation tasks [9], [21] and TD is effective for in-

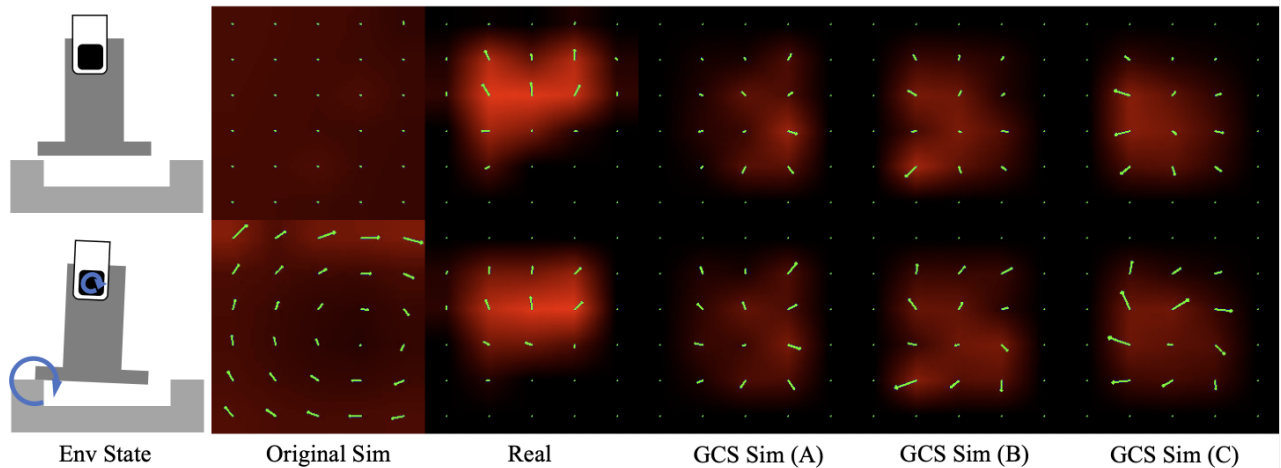


Fig. 3: The 3 rightmost columns (A-C) show GCS tactile readings from the left finger under 3 random parameters sampled from Table I. We compare them with the real-world tactile readings and the original raw simulation tactile readings under the same physical states as in Figure 1.

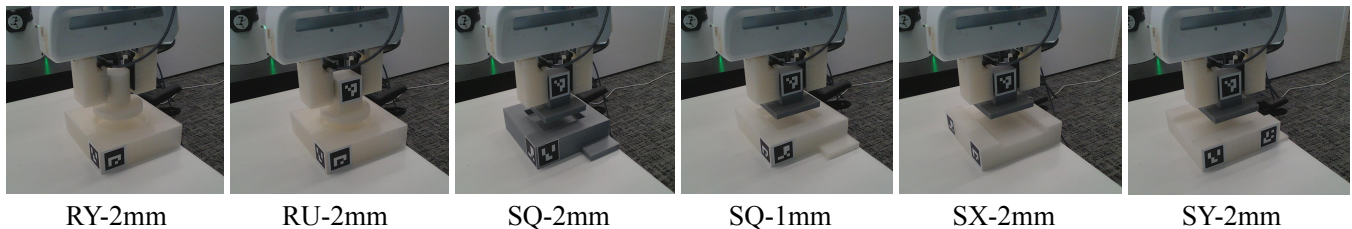


Fig. 4: Illustration of all 6 *blind* insertion tasks in our experiment.

TABLE II: 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)	0.9	0.8	0.9	0.6	0.9	0.7	0.8

hand translation [10], they are not useful for insertion tasks. In fact, the policy in TB learns to change the peg pose via extrinsic contact, in order to infer the contact with the hole base. However, such policy is very hard to transfer to real world. Additionally, in TD, the real-world shear force noise makes the feature in TD non-transferrable to the real-world rollouts. Thus, TD has a quite large sim-to-real performance gap.

With dense, distributed, 3-axes force, the policy can get $> 90\%$ success rate in simulation (DT). However, owing to large sim-to-real gap (Figure 1), the real-world success rate is low due to distribution shift.

We observe that GCS outperforms all baselines by a 50% margin on average. In many tasks such as RY-2mm, SQ-

2mm, and SX-2mm, it has a real-world success rate of 90% with barely any performance drop than in the simulation. The largest sim-to-real gap occurs in the SQ-1mm case, which is the hardest task owing to the small clearance. On SQ-1mm, one needs fine-grained and compliant control. Small action error, potentially from sim-to-real tactile gap, will cause the peg to miss the hole in the next step.

C. Ablation Study

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

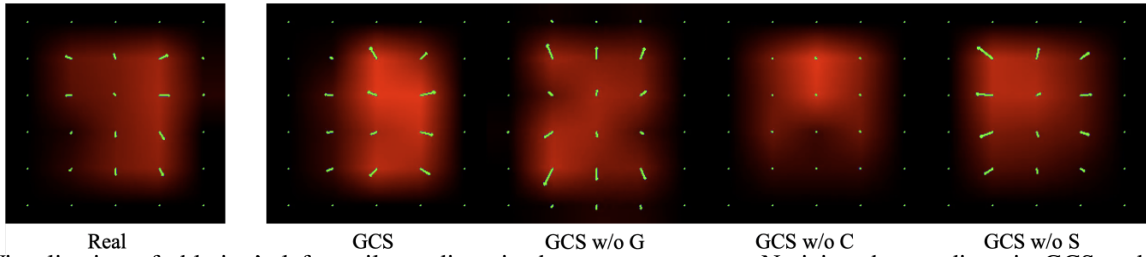


Fig. 5: Visualization of ablation’s left tactile readings in the no contact state. Noticing that readings in GCS and ablations are randomized in training.

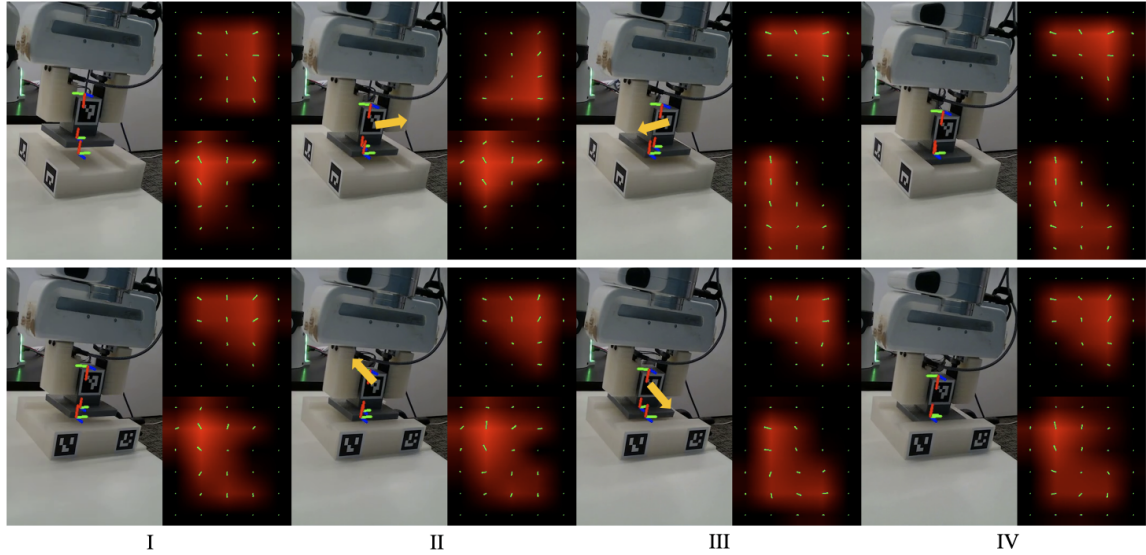


Fig. 6: Examples of policy rollout in SX-2mm (first row) and SY-2mm (second row). Stage I corresponds to the initial state, and Stage IV corresponds to the success state. Stage II and Stage III corresponds to the critical contact states that the policy infers the relative position from (Section IV-D). The yellow arrow represents the action of the current step, which can be interpreted with the tactile readings.

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 5 visualizes the tactile reading in simulation of different ablations. With all *G, C, S* techniques, the reading looks more similar to real world.

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

	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

D. Policy Interpretation

To better understand how the policy trained in simulation transfers to real world, we visualize trajectories in SX-2mm and SY-2mm tasks, in Figure 6. In SY-2mm, it detects the relative position between the peg and the hole by the change of the shear forces. In stage II, the peg contacted with the front rim and thus shear force markers on the left finger changed counterclockwise and clockwise on the right finger sensor. Consequently, the policy commanded the hand to move backward. However, it overshoot and contacted the

back rim (stage III). Again, it detected the opposite change on both sensors and thus moved forward. This time, it managed to insert the peg in the groove.

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

V. CONCLUSION

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

ACKNOWLEDGMENT

This work was partially supported by the National Science Foundation. The authors also thank Haozhi Qi for discussions on the project.

REFERENCES

- [1] W. Yuan, S. Dong, and E. H. Adelson, "Gelsight: High-resolution robot tactile sensors for estimating geometry and force," *Sensors*, vol. 17, no. 12, p. 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)*, 2022, pp. 10 781–10 787.
- [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)*. IEEE, 2021, pp. 6809–6816.
- [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*, vol. 3, no. 4, pp. 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*, vol. 40, no. 12-14, pp. 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)*. IEEE, 2023, pp. 10 408–10 414.
- [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)*. IEEE, 2023, pp. 8033–8039.
- [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*, vol. 9, no. 96, p. 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*. PMLR, 2023, pp. 2549–2564.
- [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)*. IEEE, 2016, pp. 454–460.
- [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*, vol. 3, no. 3, pp. 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. Pfennig, 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, "Eventac: 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, "Tacs1: 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] 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.
- [20] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," *arXiv preprint arXiv:2107.04034*, 2021.
- [21] 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.
- [22] T. He, C. Zhang, W. Xiao, G. He, C. Liu, and G. Shi, "Agile but safe: Learning collision-free high-speed legged locomotion," in *Robotics: Science and Systems (RSS)*, 2024.
- [23] 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*, vol. 8, no. 84, p. eadc9244, 2023. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.adc9244>
- [24] C. Zhang, W. Xiao, T. He, and G. Shi, "Wococo: Learning whole-body humanoid control with sequential contacts," 2024.
- [25] 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.
- [26] 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*. PMLR, 2023, pp. 1596–1606.
- [27] 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)*. IEEE, 2021, pp. 6437–6443.
- [28] 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)*. IEEE, 2022, pp. 10 241–10 247.
- [29] 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.
- [30] 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.
- [31] 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)*. IEEE, 2024, pp. 9234–9241.
- [32] R. Bhirangi, T. Hellebrekers, C. Majidi, and A. Gupta, "Reskin: versatile, replaceable, lasting tactile skins," in *5th Annual Conference on Robot Learning*.
- [33] 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.
- [34] 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)*. IEEE, 2024, pp. 6558–6565.
- [35] 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 visuo-tactile feedback via simulation fine-tuning," 2025.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] 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.
- [40] 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*, 2024, pp. 26 340–26 353.
- [41] 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*. IEEE, 2012, pp. 5026–5033.

- [42] Z. Si and W. Yuan, “Taxim: An example-based simulation model for gelsight tactile sensors,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 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*, vol. 7, no. 2, pp. 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*. PMLR, 2023, pp. 1488–1498.
- [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*, vol. 9, no. 3, pp. 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*, vol. 40, pp. 1509–1526, 2024.
- [48] Y. Li, W. Du, C. Yu, P. Li, Z. Zhao, T. Liu, C. Jiang, Y. Zhu, and S. Huang, “Taccel: Scaling up vision-based tactile robotics via high-performance gpu simulation,” *arXiv preprint arXiv:2504.12908*, 2025.
- [49] M. Li, Z. Ferguson, T. Schneider, T. R. Langlois, D. Zorin, D. Panozzo, C. Jiang, and D. M. Kaufman, “Incremental potential contact: intersection- and inversion-free, large-deformation dynamics,” *ACM Trans. Graph.*, vol. 39, no. 4, p. 49, 2020.
- [50] “Px-6ax gen1,” <https://paxini.com/ax/gen1>, accessed: 2025-04-23.
- [51] Y. Zhu, J. Wong, A. Mandekar, 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.
- [52] E. Coumans, “Bullet physics simulation,” in *ACM SIGGRAPH 2015 Courses*, 2015, p. 1.
- [53] 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)*. IEEE, 2024, pp. 9698–9705.
- [54] Y. Zhu, A. Joshi, P. Stone, and Y. Zhu, “Viola: Imitation learning for vision-based manipulation with object proposal priors,” *arXiv preprint arXiv:2210.11339*, 2022.
- [55] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor,” in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 80. Stockholm: Stockholmsmässan, Stockholm Sweden: PMLR, 10–15 Jul 2018, pp. 1861–1870.
- [56] “Stablebaseline3 documentation,” <https://stable-baselines3.readthedocs.io/en/master/>, accessed: 2025-04-23.
- [57] B. Tang, M. A. Lin, I. Akinola, A. Handa, G. S. Sukhatme, F. Ramos, D. Fox, and Y. Narang, “Industreal: Transferring contact-rich assembly tasks from simulation to reality,” *arXiv preprint arXiv:2305.17110*, 2023.
- [58] “Xela robotics,” <https://www.xelarobotics.com/>, accessed: 2025-09-02.

APPENDIX

Task Setup. We use OSC in all tasks. For RY-2mm, RU-2mm, SQ-2mm, SQ-1mm, the policy network outputs 3-dim position delta (XYZ) in $[-1, 1]$, scaled with 1cm, except for SQ-1mm, which is scaled with 5mm. Noticing that we freeze the rotation in OSC to simplify the tasks. For SX-2mm, we further freeze X-axis and for SY-2mm, we freeze Y-axis.

At initialization, the robot grasps the peg and moves the peg to the top of the hole. We put the bottom of the peg randomly within $[-1.5cm, 1.5cm] \times [1.5cm, 1.5cm] \times [0.5cm, 2cm]$ (XYZ) from the center of the hole.

We set a maximal 120 execution steps for SX-2mm and SY-2mm, and 200 steps for the rest of the tasks. The rollout is considered successful if the peg is successfully inserted into the hole within the trajectory length limit.

Simulation Reward. The RL reward for all tasks is defined as the weighted sum of the following 4 terms:

- *Reaching Reward.* We compute the average distance d of keypoints of the peg to correspondent points on base. The reaching reward is $r_{\text{reach}} = \frac{1}{10*d+\epsilon}$. We set weight w_{reach} to be 1.
- *Engagement Reward.* Given the engagement distance h (the distance between the bottom of peg to the base bottom), if the peg is engaged with the hole, then we have $r_{\text{engage}} = \frac{1}{10*h+\epsilon}$, otherwise $r_{\text{engage}} = 0$. We set weight w_{engage} to be 50.
- *Success Bonus.* The success bonus r_{succ} if $h < h_{\text{th}}$, where h_{th} is a threshold value. We set weight w_{succ} to be 200.
- *Action Penalty.* We penalize the action scale with $r_{\text{action}} = -\|a\|^2$. We set weight to be either w_{action} to be 0.1.

RL Policy. We select the optimal k in $\{1, 5, 10, 20\}$ for all baselines. We choose $k=1$ for TF, $k=10$ for TB, TD and DT. For GCS, we choose $k=20$ for all tasks except SY-2mm, which we find $k=10$ works better. In Section IV-C, GCS ablations use the same k as GCS. We also find that a short history has lower performance. On SQ-1mm, GCS using $k=5$ has only 30% success rate.

The tactile encoder T in TB, TD, DT, and GCS is a 4-layer CNN with BatchNorm and ReLU activation and the encoder for p_t is a single-layer MLP. All policy heads are 256-dim, 2-layer MLPs. For the critic in SAC, it is a 2-layer MLP that takes ground truth relative pose from peg to base and end-effector to base as inputs. Here, we use SAC, the off-policy algorithm, implemented in StableBaseline3 [56], for sample efficiency. Robosuite [51] is cpu-based and is relatively slow in simulation speed. Here, we use the default hyper-parameters in StableBaseline3 of SAC on all tasks.

Randomization Range. We set the randomization range of GCS (Table I) based on the readings from real-world sensors. To be specific, we use the tactile readings when the robot holds the square peg without external contact, and compared between the simulation and the real world. We choose a wide range of force scaling factors and poisson convolution coefficients $c_{x,y}, \alpha_k, \beta_{ij}$, to cover the reference readings in absolute scale. We set $s_{x,y}$ and $h_{\text{max,min}}$ in the simulator to generate a similar normal force distribution. Quantitatively, we compute the MSE between the simulation forces and real tactile forces, under the no contact state when holding the square peg. Averaging over all taxels, the original simulation tactile readings have a MSE of 0.68 while the smallest MSE with GCS over 5 random instances is only 0.11.

Sensor Calibration. It is inconvenient to calibrate magnet-based tactile sensors, which requires specialized hardware [12], [13]. Moreover, manufacturers [58] will charge a significant amount of money for per-device sensor calibration. Our GCS allows sim-to-real transfer without requiring sensor calibration process and instead learns a robust policy with large randomization.