

Sim2Real Manipulation on Unknown Objects with Tactile-based Reinforcement Learning

Entong Su¹, Chengzhe Jia^{1,†}, Yuzhe Qin^{1,†}, Wenxuan Zhou^{2,†}, Annabella Macaluso^{1,†}, Binghao Huang³, Xiaolong Wang¹

Abstract—Using tactile sensors for manipulation remains one of the most challenging problems in robotics. At the heart of these challenges is generalization: How can we train a tactile-based policy that can manipulate unseen and diverse objects? In this paper, we propose to perform Reinforcement Learning with only visual tactile sensing inputs on diverse objects in a physical simulator. By training with diverse objects in simulation, it enables the policy to generalize to unseen objects. However, leveraging simulation introduces the Sim2Real transfer problem. To mitigate this problem, we study different tactile representations and evaluate how each affects real-robot manipulation results after transfer. We conduct our experiments on diverse real-world objects and show significant improvements over baselines. Our project page is available at <https://tactilerl.github.io/>.

I. INTRODUCTION

When unlocking a door, we may reach into our bags for a key and re-orient it before inserting it into the lock to open the door; this series of actions are accomplished with a strong reliance on tactile cues. To enable robots to obtain similar skills, different tactile sensors [1]–[4] have been designed and shown to be effective in capturing normal and shear forces for various manipulation tasks, especially when visual information is occluded. Following these works, the recent design of visual-tactile sensors [5]–[7] has further improved the resolution and sensitivity for capturing rich contact information and is more friendly to use with learning algorithms [8]–[13]. In this paper, we aim to explore the use of tactile sensors for the pivoting task as visualized in Figure 1. The main challenge of the task is to understand the pose and geometry of the object and act accordingly based on tactile information at the fingertip of the robot gripper.

One straightforward approach is to use tactile information to first estimate the object shape and pose, and directly use the estimated information as policy input [14]–[20]. However, such estimations will likely be inaccurate given only partial observations, which will introduce errors in the policy execution. Instead of explicit estimation of object geometry, we can leverage end-to-end Reinforcement Learning (RL), which takes the visual tactile observations as inputs, and directly outputs the action for robust manipulation [21]–[24]. However, executing RL policies on real robots requires a significant number of interactions, which increases along

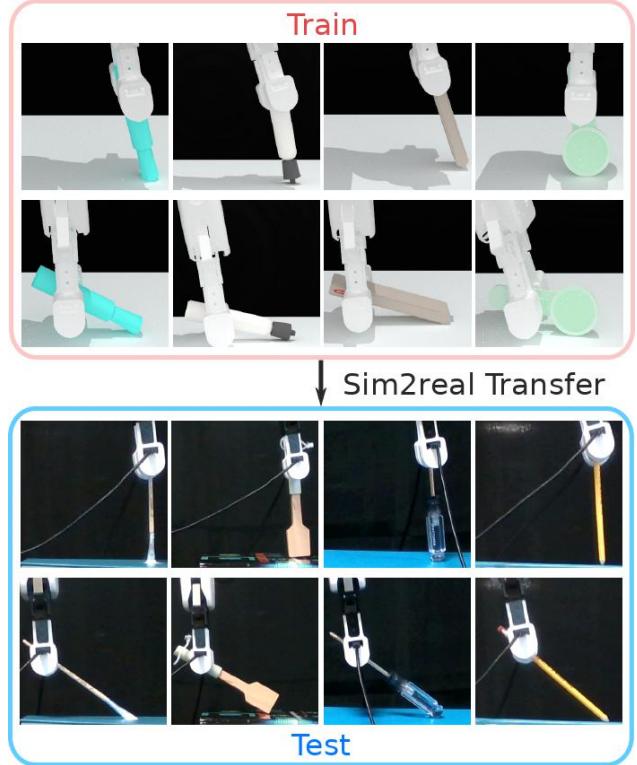


Fig. 1: We study the task of pivoting an object to a target angle with only tactile observations. Our tactile-based policy, trained with Reinforcement Learning purely in simulation, successfully transfers to the real robot without real-world data. The first row in each block visualizes the initial state of different episodes, while the second row demonstrates the final execution results.

with the complexity of the task. This prevents the policy from learning and operating with diverse objects.

In this paper, we propose a system to train an RL policy with tactile inputs in simulation with diverse objects and then perform zero-shot Sim2Real transfer to the real robot, as shown in Figure 1. By leveraging simulation, we largely increase the number and diversity of objects the robot interacts with, which leads to a more robust policy. However, there is still a large Sim2Real gap in how the tactile image is formed between the simulator and the real-world tactile sensor. Instead of performing perfect alignment between sim and real sensor imaging, we propose to study how to extract sufficient information from the sensor and reduce the domain

¹ University of California San Diego, CA, USA

² Carnegie Mellon University, PA, USA

³ University of Illinois Urbana-Champaign, IL, USA

† These authors contributed equally.

gap under the context of RL inputs. Specifically, we ablate the tactile inputs by using: (i) the original tactile reading from the sensor; (ii) the difference of tactile information between the current reading and a canonical reading with no force applied; (iii) binary tactile information processed from the original tactile reading. The final version of observation provides less object information but turns out to reduce the Sim2Real gap effectively.

We experiment with different tactile representations on the tasks of pivoting [25], [26]. Different from previous works that experiment with limited objects, we perform training and testing on a large number of objects, including 22 training objects in the simulation and 16 unseen objects for testing in the real world. By leveraging diverse training objects, the policy achieves better real-robot performance on a more abstract level of tactile representation compared to the full tactile information. To the best of our knowledge, our work is the first to conduct Sim2Real transfer and achieve diverse object generalization for tactile-based manipulation.

II. RELATED WORK

Vision-based Tactile Sensing. In recent years, vision-based tactile sensing has gained significant attention to enhance robot interaction with the environment. These sensors offer many benefits over force-based tactile sensors, such as increased spatial resolution and more detailed contact geometry. GelSight [1] is a pioneering work in this field, utilizing elastomeric material and a light-camera system to capture contact geometry. Following this, researchers have developed various methods to improve the design of such sensors, including OmniTact [7], GelSlim [5], TacTip [2], GelTip [6], and Digit [27]. To make use of the high-resolution data from these sensors, previous methods first build a state estimator from the tactile image and then integrate it with model-based control. For example, Oller *et al.* [28] uses Iterative Closest Points to estimate the object pose from contact points for down-stream manipulation task; Wilson *et al.* [20] and She *et al.* [14] use Principal Component Analysis to estimate the orientation of contact shape from GelSight reading for cable routing task. These methods encode the information from high-resolution images as low-dimensional states, which does not fully capitalize on the detailed contact geometry. To better leverage the rich sensory data provided by vision-based tactile sensors, researchers employ end-to-end reinforcement learning for directly mapping tactile images to robot actions [29]–[32]. However, the dynamics of tactile sensing are guided by complex contact mechanics, leading to a significant Sim2Real gap. The work most closely related to ours is by Kim *et al.* [15], who achieved the pivoting task by estimating the contact displacement between the gripper and the object. However, their method is performed directly in the real world, making it difficult to scale up to a wide variety of objects. In contrast, our approach, which is trained purely in simulation using diverse objects, can be transferred to the real world to manipulate unseen objects across multiple categories. We achieve this by focusing on extracting sufficient information from tactile images.

Sim2Real Transfer for Tactile Sensing. Recently, several works have focused on bridging the gap between tactile simulation and real-world sensors. Yin *et al.* [33] proposed binarizing the Force Sensing Resistor (FSR) sensor signal to address in-hand rotating tasks, while Liang *et al.* [34] and Hebert *et al.* [35] utilized binary contact modes to transfer policies on BioTac sensors for object pose tracking. These approaches effectively minimize the Sim2Real gap by converting both modalities into the binary domain. However, the process of binarizing tactile signals for vision-based tactile sensors with high spatial resolution remains unclear. To obtain more realistic and dense tactile forces, recent work [36]–[39] combined ‘Finite Element Modeling’ (FEM) [40] and learning-based methods to simulate tactile sensor deformations. Although these simulators yield highly accurate tactile readings, their computational cost is prohibitively high, rendering them unsuitable for reinforcement learning (RL) training. To enhance simulation efficiency, recent work [41]–[43] train CycleGAN [44] on a self-collected dataset to convert tactile images across different domains. On the other side, Xu *et al.* [45] introduced a penalty-based model featuring differentiable tactile simulation. While their method demonstrates promising Sim2Real transfer for robot manipulation, it has not been shown to generalize effectively to diverse and novel objects in the real world. Rather than aligning sim and real sensor imaging with better simulation, our approach explores how various tactile image representations can reduce the domain gap. Moreover, our method does not necessitate real-world data, thereby eliminating the need for additional data collection efforts.

III. SIM2REAL TRANSFER OF VISUAL TACTILE READINGS

We propose a system designed for Sim2Real transfer of RL policies with tactile observations from the DIGIT tactile sensors (refer to Fig. 2). Notably, the proposed system does not depend on any real-world data. In this section, we will first explain how we generate visual-tactile data in the simulator to learn the pivoting task. Furthermore, to bridge the Sim2Real domain gap, we investigate three distinct representations to encode the tactile data.

A. Tactile Pattern Rendering in Simulation

In order to simulate the pattern of the DIGIT sensor, we extend the SAPIEN [46], [47] simulator to accommodate visual tactile sensors, adhering to a pipeline similar to that in TACTO [48]. Three light sources are initialized, and the gel mesh is configured to align with the physical design of DIGIT. We update tactile images in real time using a linear mapping approach to convert contact forces from the physics engine into deformation depth. This mapping enables object position adjustments based on applied normal forces. Then, we render RGB tactile images using Phong’s model [49]. To speed up RL training, we replace TACTO’s original OpenGL renderer with PyTorch3D. This modification enables direct GPU tensor rendering, thereby eliminating the overhead for GPU-CPU data transfer.

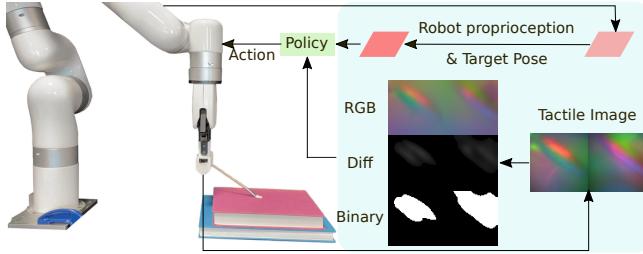


Fig. 2: Setup. Two tactile sensors are mounted on the gripper’s fingertips. Tactile readings are processed using our proposed approach and then serve as the input for training RL policies.

B. Tactile Image Representation

Performing RL with tactile input within a Sim2Real pipeline presents two main challenges. First, accurately simulating the visual-tactile sensor to mirror its real-world counterpart is difficult due to the Sim2Real gap. (ii) Real-world DIGIT sensors behave inconsistently due to the variations in manufacturing processes and gel properties, further complicating the alignment between simulated and real sensors. Therefore, instead of focusing on achieving perfect alignment in simulation, we investigate tactile image representations that can abstract the non-essential details in tactile images while preserving sufficient contact information that is crucial for robot decision-making.

Our study explores three tactile image representations, as illustrated in Figure 2. The first representation is the original tactile RGB image directly acquired from the tactile simulation, as discussed in Section III-A (denoted as **RGB**). Secondly, we propose a representation called **Diff**, which subtracts the current image from a force-free canonical image. This is done by calculating pixel-wise differences between the two RGB images and converting them to grayscale by averaging the RGB values. Thirdly, we explore a binary version of the **Diff** image, which is denoted as **Binary**. This involves applying a predefined threshold (ϕ) to distinguish between contacted and non-contacted pixels. We conduct a grid search to tailor the ϕ threshold individually for different digits to accommodate DIGIT’s manufacturing variations and minimize noise. However, there is a trade-off between noise reduction and the possibility of missing certain contact information, as shown in Figure 5.

We augment the **Diff** and **Binary** tactile images by randomly scaling them between 0 and 1. To align the data between the left and right grippers, we horizontally flip the tactile image from the right gripper to correspond with the image from the left gripper. These tactile representations are used both in simulation and real-world experiments.

IV. LEARNING TACTILE POLICIES FOR PIVOTING

This section discusses how our proposed system uses tactile readings to train RL policies for the pivoting task, as shown in Figure 1.

Task Definition: In the pivoting task, the robot needs to rotate the object to a target angle relative to the robot gripper

pose. This operation exclusively depends on tactile sensing and joint proprioception, without any need for external sensors to estimate the object’s pose.

Observation Space: The observation space contains tactile images, robot joint proprioceptive states, and task-related information (e.g., target angle). As mentioned in Section III-B, we examine three different representations of tactile images, each with a resolution of 64×64 .

Action Space: Since the focus of this work is on the robotic system rather than verifying the RL algorithm, we design the action space to be as simple as possible to speed up RL training for this task. We restrict the end effector translation to the xz plane and restrict rotation to the y-axis. The action space does not include the gripper width, as it is commanded to grasp the object securely.

Domain Randomization: We use diverse objects from PartNet [50] and Breaking Bad [51]. The objects used for the simulation training and real evaluation are shown in Figure 3a. We randomize the height of supporting surface from 0 to 20 cm relative to the robot base. The object length ranges from 13 to 18 cm, and its initial pose varies between 165 and 195 degrees relative to the gripper. The target relative angle is randomized between 90 and 150 degrees.

Policy Training: We use the Proximal Policy Optimization (PPO) [52] for RL training. Two images are encoded using shared encoders and combined with the MLP feature of proprioception states. Both the actor and critic networks utilize the same feature. We use five seeds for the training. We use the default hyperparameter in [stable-baselines3](#) for the training.

Reward Function: The reward functions comprise four components: contact, distance-based, angle-based, and action penalties. The reward function is as follows:

$$R = w_{contact}r_{contact} + w_{position}r_{position} + w_{angle}r_{angle} - w_{penalty}r_{penalty}$$

- **Contact:** Gripper gets $+0.5$ reward ($r_{contact}$) when it interacts with the objects. $w_{contact}$ can be 0, 1, 2 based on the number of tactile sensors in contact with the object.
- **Distance-based reward:** Objects are rewarded for approaching the target position and penalized for moving away. $w_{position} = 10$ is used when a gripper contacts the object, and $r_{position}$ ranges from -1 to 1, calculated as $r_{position} = 1 - (cur_{dist}/init_{dist})$.
- **Angle-based reward:** This term is given by the difference between the current and target angles, following a structure similar to the distance-based reward.
- **Action penalty:** This term penalizes the magnitude of action output, defined as $r_{penalty} = ||a||^2$ with $w_{penalty} = 0.01$.

Baselines: We compared our proposed method with several baselines ranging from angle estimation, visual feedback, and expert demonstration. The baselines are categorized as follows:

- 1) w/o Tactile: Only using proprioception observation.
- 2) Oracle Angle: Trained with ground truth angles.

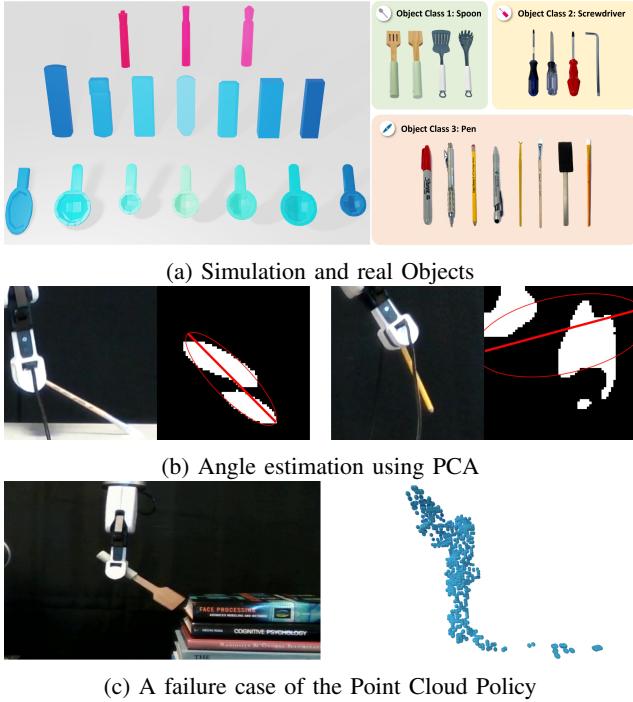


Fig. 3: Object categories, Visualization of angle estimation using PCA and failure case of PC policy. We use diverse objects(a) for simulation training(left image) and real evaluation(right image). For (b), we show the cases of PCA angle estimation. Right: a successful example. Left: a failure case. The red line represents the estimated orientation from PCA. For (c), we show the failure case of the Point Cloud policy.

- 3) Angle Estimator: We use ConvNeXt [53] for object-in-hand pose estimation by using the tactile Binary image. The estimated angle is used for the Oracle Angle policy in the real robot experiment.
- 4) PCA angle: PCA is employed to predict object orientation from tactile images, which is used on the Oracle angle policy.
- 5) Point Cloud: We utilize point cloud data as observation for policy training, excluding the supporting table, and employ the PointNet architecture as the backbone.
- 6) DAgger: The Tactile-Binary (Aug) policy guides the student policy, which doesn't rely on tactile information.
- 7) Tactile-RGB, Tactile-Depth, Tactile-Diff (**Ours**): We utilize tactile readings without employing image augmentation during training.
- 8) Tactile-RGB (Aug), Tactile-Depth (Aug), Tactile-Diff (Aug) (**Ours**): We enhance our methods to incorporate image augmentation for tactile images, including the operations of random scale, erase. We introduce additional adjustments for RGB images on the brightness, contrast, and color hues.

Evaluation Metric: We evaluate our policy based on two metrics: angle deviation and success rate. Angle deviation is the difference between the current and target rotated angles, expressed as a ratio. A task is considered successful when

the angle deviation is under 15%. Evaluation is based on the average success rate/error of these five seeds in the simulation. We evaluated 30 episodes in total in the real experiment.

V. SIM2REAL TRANSFER OF VISUAL TACTILE READINGS

We conduct Sim2Real transfer experiments on the real robot to evaluate the effectiveness of our policy, as shown in Figure 5. We directly transfer our policy to the real robot without any fine-tuning. In this section, we mainly investigate three aspects: 1) The necessity of tactile sensing, 2) The overall performance by using different tactile representations, and 3) The generalization of multi-category tactile policies.

A. Necessity of Tactile Sensing

To assess the importance of tactile sensing, we compare our methods with various baseline approaches described in Section IV. These baselines include observations such as angle estimation, visual feedback, and expert demonstration. Figure 4 and Table I summarize the training curves and evaluation results.

Comparison to w/o tactile: We established a baseline for the tasks without tactile feedback. Compared with other methods, this baseline exhibited poorer performance, with lower success rates and larger angle deviations (first row in Table I). Figure 4 shows that the policy without tactile feedback learns slowly and shows a higher variance in reward and success rate. This highlights the importance of visual feedback, state information, and tactile feedback for the pivoting.

Comparison Against angle estimation methods: As shown in Table I, the Oracle Angle policy achieves a 0.96 success rate in simulation, but this policy encounters the challenge of Sim2Real transfer. Using PCA and ConvNeXt for angle estimation achieves lower real-world success rates of 0.23 and 0.60, while Tactile-RGB(Aug) and Tactile-Binary(Aug) policies achieve approximately 0.80. Figure 3b illustrates the failure case of the PCA method. This gap indicates real-world factors, like noise, impacting angle estimation precision and task success.

Comparison to Point Cloud policy: We also compared our policy with the policy solely relying on visual feedback. As shown in Table I, the Point Cloud policy's success rate drops from 0.62 in simulation to 0.50 in the real world. Figure 3c shows the failure case of this policy. The main reason for these failures is the significant differences in size and shape between real-world objects and those in the simulation.

Comparison to DAgger: We used the tactile-binary policy, with a simulation success rate of 0.69, as the expert policy to guide the student policy without tactile information. The student policy achieved a similar success rate to the teacher policy in the simulation, but its real-world performance decreased to 0.50. This highlights the crucial role of tactile sensing in learning and decision-making during manipulation.

Environment Method	Simulation		Real-World	
	Deviation ↓	Success ↑	Deviation ↓	Success ↑
w/o Tactile	34.84% ± 35.38%	0.32 ± 0.23	23.55% ± 8.00%	0.33 ± 0.09
DAgger	17.14% ± 3.04%	0.66 ± 0.05	18.05% ± 10.04%	0.50 ± 0.09
Angle Estimator	—	—	21.66% ± 10.98%	0.60 ± 0.08
Point Cloud	16.2% ± 2.14%	0.62 ± 0.010	19.32% ± 2.14%	0.50 ± 0.11
PCA Angle	—	—	30.19% ± 14.32%	0.23 ± 0.14
<i>Oracle Angle</i>	9.22% ± 1.43%	0.96 ± 0.02	—	—
Tactile - RGB	14.34% ± 2.22%	0.64 ± 0.10	18.03% ± 7.42%	0.50 ± 0.14
Tactile - Diff	15.67% ± 3.33%	0.67 ± 0.10	16.04% ± 4.00%	0.60 ± 0.04
Tactile - Binary	15.31% ± 1.21%	0.65 ± 0.05	12.25% ± 3.70%	0.80 ± 0.04
Tactile - RGB(Aug)	13.19% ± 2.77%	0.75 ± 0.05	11.56% ± 6.44%	0.76 ± 0.06
Tactile - Diff(Aug)	14.35% ± 2.49%	0.67 ± 0.03	14.51% ± 3.78%	0.66 ± 0.09
Tactile - Binary(Aug)	12.98% ± 2.47%	0.69 ± 0.07	11.15% ± 3.34%	0.80 ± 0.02

TABLE I: **Comparison of different observation modalities and tactile representations.** We evaluate the angle deviation and success rate for policy trained with three tactile image representations (RGB, Diff., Binary), visual feedback (Point Cloud), expert demonstration (DAgger), angle information (Oracle Angle, PCA Angle, Angle Estimator), as well as without tactile (w/o Tactile) sensing in both simulated and real environments. Please note that the addition of (**Aug**) indicates that the tactile representations have undergone image augmentation, and Oracle Angle is used as an upper bound only in the simulation experiment.

Test Object type	Method	Simulation		Real-World		
		Observations	Objects	Deviation ↓	Success ↑	
Single Category	w/o Tactile	Single Category	24.45% ± 14.37%	0.52 ± 0.37	26.68% ± 8.01%	0.36 ± 0.09
	Tactile - Binary(Aug) (Ours)	Single Category	8.23% ± 1.14%	0.91 ± 0.02	14.03% ± 4.60%	0.54 ± 0.07
Multi Category	w/o Tactile	Single Category	72.12% ± 11.58%	0.30 ± 0.05	23.55% ± 8.0%	0.33 ± 0.16
	Tactile - Binary(Aug) (Ours)	Single Category	46.71% ± 13.99%	0.42 ± 0.13	21.72% ± 6%	0.53 ± 0.19
Multi Category	w/o Tactile	Multi Category	34.84% ± 35.38%	0.32 ± 0.23	23.55% ± 8.00%	0.33 ± 0.09
	Tactile - Binary(Aug) (Ours)	Multi Category	12.98% ± 2.47%	0.69 ± 0.07	11.15% ± 3.34%	0.80 ± 0.08

TABLE II: **Effect of multi objects training in simulation and real-world.** We contrast evaluation results for single and multi-object category training in pivoting tasks based on Tactile-Binary(Aug) and w/o Tactile policy.

Method	Solid Table		Soft Table	
	Deviation ↓	Success ↑	Deviation ↓	Success ↑
w/o Tactile	23.55% ± 8.00%	0.33 ± 0.09	22.26% ± 11.70%	0.21 ± 0.14
Tactile - Binary (Ours)	12.25% ± 3.7%	0.80 ± 0.04	11.60% ± 3.83%	0.76 ± 0.09

TABLE III: **Generalization to unseen supporting surfaces in the real world.** We summarize the angle deviation ratio and Real-World Pivoting Experiment success rate on different types of surfaces.

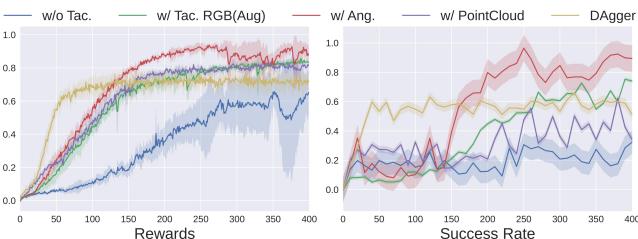


Fig. 4: **Training curves.** We report the training curves for each task with two metrics: reward and success rate in the simulation. Given that the tactile policies exhibit similar reward and success rate trends, we present the results for Tac. RGB (Aug) for simplicity. Note that *Oracle Angle* achieves the best performance because it uses the ground truth object pose, which serves as an upper bound in the simulation experiment.

Evaluation Variation: We notice that the tactile policy achieves a higher real-world success rate than simulations, which can be partially attributed to the differences in the evaluation setup. We evaluate 500 episodes in the simulation while evaluating 30 episodes in the real. In simulations, we test in a more diverse environment with different types of objects than in the real world.

In summary, our tactile-based methods outperformed all others with the highest real-world success rate and the lowest deviation in the real-world evaluation. This result demonstrates the crucial role of tactile sensing within our system.

B. Effect of Tactile Representations

In this section, we explore the impact of policy training by utilizing different tactile representations(RGB, Binary, and Difference) and image augmentation on Sim2Real transfer.

The evaluation results are summarized in Table I.

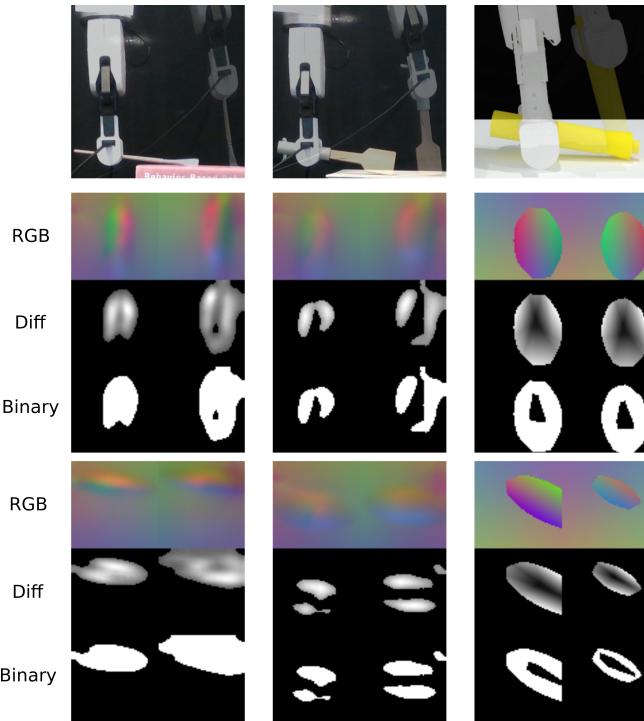


Fig. 5: Real and Sim Experiment for pivoting. We assessed our pivoting task policy using tactile-based policies with RGB, difference, and binary representations. The left two and third columns show the initial and final states of pivoting in the real robot and simulation, respectively. The second and third rows depict the tactile images from the first and last frames captured by two grippers. The bottom two rows illustrate the RGB image (RGB), difference image (Diff), and binary image (Binary) from top to bottom, respectively.

Effect of different Tactile representations: We compared different tactile representations without using image augmentation(Table I (lines 7 to 9)). Despite all three policies achieving roughly 0.65 success rates with a 15% angle deviation in simulations, we show that they exhibit distinct behaviors in real-world robotic experiments. The Tactile-RGB policy performs less than other tactile-based policies in real evaluation. In contrast, Tactile-Binary policies consistently demonstrated superior performance in these real-world experiments. This indicates the challenges of adapting RGB tactile representations for lighting, color, and pixel value variations during sim2real transfer.

Effect of Image augmentation: In Section IV, we applied image augmentation on tactile representations and summarized the evaluation results in Table I (lines 10 to 12). Compared with the non-augmentation one, The tactile-RGB policy with image augmentation shows substantial improvements, reaching a success rate of 0.75 in simulation and 0.76 in the real world. The Tactile-Binary and Tactile-Diff policies also exhibited slight performance enhancements in both simulation and real life. These results illustrate that

image augmentation can enhance the adaptability of tactile images for Sim2Real transfer.

In summary, employing image augmentation can yield significant benefits for Sim2Real transfer, particularly when dealing with RGB images. This is because image augmentation helps policies focus more on contact patterns rather than pixel values during training. As a result, it enhances the performance of policies relying on these three representations, resulting in more favorable outcomes.

C. Multi-Category Tactile Policy Generalization

In this section, we explore how multi-object training improves the generalizability when evaluating multiple objects and previously unseen supporting surfaces.

Effect of Multi-Objects Training: We compared policies trained on single or multiple object categories for the pivoting task and the result is summarized in Table II. Both training datasets contain multiple instances of each object category. The performance of single-category policy has significantly declined in real-world evaluation. In contrast, the multi-category tactile policy demonstrates superior generalization capabilities with a success rate of 0.80.

Generalization to unseen supporting surfaces: We also evaluated how well the policies can adapt to varying surfaces with different friction and stiffness properties. We evaluated this by using Tactile-RGB, Tactile-Binary, and Tactile-Binary policies. Each policy evaluated ten trials to ensure robustness and consistency, with results summarized in Table III. Despite a performance drop in both tactile-based and non-tactile methods, the tactile-based policy maintained a success rate of 0.76 with a 12% angle deviation. This highlights the tactile-based policy can effectively adapt to various environmental conditions.

Failure Cases: There are two primary factors contributing to the failures. The first is unstable gripping, leading to incomplete patterns. The second is incomplete or unusual contact, resulting in poor tactile feedback. In summary, the result above demonstrates that our tactile-based policies are generalized to multi-categories and perform well on the previously unseen supporting surface in the real evaluation.

VI. CONCLUSION

In our research, we developed and trained a tactile-focused reinforcement learning policy tailored for the task of pivoting. We then successfully executed a transfer from simulation to the real world. Our findings indicate that our approach is versatile, effectively adapting to a wide range of unfamiliar objects and surfaces not encountered during training. Future work could include integrating tactile feedback to refine the manipulation strategy.

VII. ACKNOWLEDGEMENT

Acknowledgment: This work was supported, in part, by the Qualcomm Innovation Fellowship, and the Technology Innovation Program (20018112, Development of autonomous manipulation and gripping technology using imitation learning based on visual and tactile sensing) funded by the Ministry of Trade, Industry & Energy (MOTIE), Korea.

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] B. Ward-Cherrier, N. Pestell, L. Cramphorn, B. Winstone, M. E. Giannaccini, J. Rossiter, and N. F. Lepora, "The tactip family: Soft optical tactile sensors with 3d-printed biomimetic morphologies," *Soft robotics*, vol. 5, no. 2, pp. 216–227, 2018.
- [3] A. Alspach, K. Hashimoto, N. Kuppuswamy, and R. Tedrake, "Soft-bubble: A highly compliant dense geometry tactile sensor for robot manipulation," 2019.
- [4] C. Sferrazza and R. D'Andrea, "Design, motivation and evaluation of a full-resolution optical tactile sensor," *Sensors*, vol. 19, no. 4, p. 928, 2019.
- [5] I. Taylor, S. Dong, and A. Rodriguez, "Gelslim3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger," 2021.
- [6] D. F. Gomes, Z. Lin, and S. Luo, "Geltip: A finger-shaped optical tactile sensor for robotic manipulation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 9903–9909.
- [7] A. Padmanabha, F. Ebert, S. Tian, R. Calandra, C. Finn, and S. Levine, "Omnitact: A multi-directional high-resolution touch sensor," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 618–624.
- [8] M. B. Villalonga, A. Rodriguez, B. Lim, E. Valls, and T. Sechopoulos, "Tactile object pose estimation from the first touch with geometric contact rendering," in *Conference on Robot Learning*. PMLR, 2021, pp. 1015–1029.
- [9] S. Zhong, A. Albini, O. P. Jones, P. Maiolino, and I. Posner, "Touching a neRF: Leveraging neural radiance fields for tactile sensory data generation," in *6th Annual Conference on Robot Learning*, 2022. [Online]. Available: <https://openreview.net/forum?id=No3mbanRIZJ>
- [10] Y. Zhang, W. Yuan, Z. Kan, and M. Y. Wang, "Towards learning to detect and predict contact events on vision-based tactile sensors," in *Conference on Robot Learning*. PMLR, 2020, pp. 1395–1404.
- [11] C. Higuera, B. Boots, and M. Mukadam, "Learning to read braille: Bridging the tactile reality gap with diffusion models," 2023.
- [12] Y. Zhao, X. Jing, K. Qian, D. F. Gomes, and S. Luo, "Skill generalization of tubular object manipulation with tactile sensing and sim2real learning," *Robotics and Autonomous Systems*, vol. 160, p. 104321, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S092188902200210X>
- [13] V. Kakani, X. Cui, M. Ma, and H. Kim, "Vision-based tactile sensor mechanism for the estimation of contact position and force distribution using deep learning," *Sensors*, vol. 21, no. 5, p. 1920, 2021.
- [14] 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. [Online]. Available: <https://doi.org/10.1177/02783649211027233>
- [15] S. Kim, D. K. Jha, D. Romeres, P. Patre, and A. Rodriguez, "Simultaneous tactile estimation and control of extrinsic contact," 2023.
- [16] M. Bauza, E. Valls, B. Lim, T. Sechopoulos, and A. Rodriguez, "Tactile object pose estimation from the first touch with geometric contact rendering," 2020.
- [17] R. Gao, Y.-Y. Chang, S. Mall, L. Fei-Fei, and J. Wu, "Objectfolder: A dataset of objects with implicit visual, auditory, and tactile representations," *arXiv preprint arXiv:2109.07991*, 2021.
- [18] N. Kuppuswamy, A. Castro, C. Phillips-Grafflin, A. Alspach, and R. Tedrake, "Fast model-based contact patch and pose estimation for highly deformable dense-geometry tactile sensors," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1811–1818, 2020.
- [19] N. Sunil, S. Wang, Y. She, E. Adelson, and A. R. Garcia, "Visuotactile affordances for cloth manipulation with local control," in *Proceedings of The 6th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, K. Liu, D. Kulic, and J. Ichnowski, Eds., vol. 205. PMLR, 14–18 Dec 2023, pp. 1596–1606. [Online]. Available: <https://proceedings.mlr.press/v205/sunil23a.html>
- [20] A. Wilson, H. Jiang, W. Lian, and W. Yuan, "Cable routing and assembly using tactile-driven motion primitives," 2023.
- [21] J. Hansen, F. Hogan, D. Rivkin, D. Meger, M. Jenkin, and G. Dudek, "Visuotactile-rl: Learning multimodal manipulation policies with deep reinforcement learning," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 8298–8304.
- [22] A. Church, J. Lloyd, N. F. Lepora *et al.*, "Tactile sim-to-real policy transfer via real-to-sim image translation," in *Conference on Robot Learning*. PMLR, 2022, pp. 1645–1654.
- [23] T. Bi, C. Sferrazza, and R. D'Andrea, "Zero-shot sim-to-real transfer of tactile control policies for aggressive swing-up manipulation," 2021.
- [24] 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," *arXiv preprint arXiv:2312.01853*, 2023.
- [25] Y. Hou, Z. Jia, and M. T. Mason, "Fast planning for 3d any-pose-reorienting using pivoting," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1631–1638.
- [26] ———, "Reorienting objects in 3d space using pivoting," *arXiv preprint arXiv:1912.02752*, 2019.
- [27] M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer, D. Jayaraman, and R. Calandra, "DIGIT: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 3838–3845, jul 2020. [Online]. Available: <https://doi.org/10.1109/LFlra.2020.2977257>
- [28] M. Oller, M. Planas, D. Berenson, and N. Fazeli, "Manipulation via membranes: High-resolution and highly deformable tactile sensing and control," 2022.
- [29] 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)*, 2021, pp. 6437–6443.
- [30] H. Li, Y. Zhang, J. Zhu, S. Wang, M. A. Lee, H. Xu, E. Adelson, L. Fei-Fei, R. Gao, and J. Wu, "See, hear, and feel: Smart sensory fusion for robotic manipulation," 2022.
- [31] H. Van Hoof, T. Hermans, G. Neumann, and J. Peters, "Learning robot in-hand manipulation with tactile features," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2015, pp. 121–127.
- [32] H. Xu, Y. Luo, S. Wang, T. Darrell, and R. Calandra, "Towards learning to play piano with dexterous hands and touch," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 10410–10416.
- [33] 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.
- [34] J. Liang, A. Handa, K. Van Wyk, V. Makovychuk, O. Kroemer, and D. Fox, "In-hand object pose tracking via contact feedback and gpu-accelerated robotic simulation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6203–6209.
- [35] P. Hebert, N. Hudson, J. Ma, and J. Burdick, "Fusion of stereo vision, force-torque, and joint sensors for estimation of in-hand object location," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 5935–5941.
- [36] Y. S. Narang, B. Sundaralingam, K. V. Wyk, A. Mousavian, and D. Fox, "Interpreting and predicting tactile signals for the syntouch biotac," 2021.
- [37] Y. Narang, B. Sundaralingam, M. Macklin, A. Mousavian, and D. Fox, "Sim-to-real for robotic tactile sensing via physics-based simulation and learned latent projections," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6444–6451.
- [38] Z. Ding, N. F. Lepora, and E. Johns, "Sim-to-real transfer for optical tactile sensing," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1639–1645.
- [39] D. Ma, E. Donlon, S. Dong, and A. Rodriguez, "Dense tactile force estimation using gelslim and inverse fem," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 5418–5424.
- [40] C. C. Spyros, *Finite Element Modeling*. West Virginia Univ. Press Morgantown, WV, USA, 1994.
- [41] W. Chen, Y. Xu, Z. Chen, P. Zeng, R. Dang, R. Chen, and J. Xu, "Bidirectional sim-to-real transfer for gelsight tactile sensors with cyclegan," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6187–6194, 2022.
- [42] T. Jianu, D. F. Gomes, and S. Luo, "Reducing tactile sim2real domain gaps via deep texture generation networks," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 8305–8311.

- [43] Y. Zhao, X. Jing, K. Qian, D. F. Gomes, and S. Luo, "Skill generalization of tubular object manipulation with tactile sensing and sim2real learning," *Robotics and Autonomous Systems*, vol. 160, p. 104321, 2023.
- [44] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.
- [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 *6th Annual Conference on Robot Learning*, 2022. [Online]. Available: <https://openreview.net/forum?id=6BIffCl6gsM>
- [46] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang, L. Yi, A. X. Chang, L. J. Guibas, and H. Su, "SAPIEN: A simulated part-based interactive environment," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [47] X. Zhang, R. Chen, A. Li, F. Xiang, Y. Qin, J. Gu, Z. Ling, M. Liu, P. Zeng, S. Han *et al.*, "Close the optical sensing domain gap by physics-grounded active stereo sensor simulation," *IEEE Transactions on Robotics*, 2023.
- [48] 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, apr 2022. [Online]. Available: <https://doi.org/10.1109%2Flra.2022.3146945>
- [49] K. Genova, F. Cole, A. Maschinot, A. Sarna, D. Vlasic, and W. T. Freeman, "Unsupervised training for 3d morphable model regression," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [50] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su, "Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding," 2018.
- [51] S. Sellán, Y.-C. Chen, Z. Wu, A. Garg, and A. Jacobson, "Breaking bad: A dataset for geometric fracture and reassembly," 2022.
- [52] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017.
- [53] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," 2022.