Touch Insight Acquisition: Human's Insertion Strategies Learned by Multi-Modal Tactile Feedback

Kelin Yu[1], **Yunhai Han**[1], **Matthew Zhu**[1], **Ye Zhao**[1]
[1]**Georgia Institute of Technology**
(kyu85, yhan389, matthewjzhu0520, yezhao)@gatech.edu

Abstract

In the evolving landscape of robotics and automation, the application of touch processing is crucial, particularly in learning to execute intricate tasks like insertion. However, existing works focusing on tactile methods for insertion tasks predominantly rely on sensor data and do not utilize the rich insights provided by human tactile feedback. For utilizing human sensations, methodologies related to learning from humans predominantly leverage visual feedback, often overlooking the invaluable tactile insights that humans inherently employ to finish complex manipulations. Addressing this gap, we introduce "MimicTouch", a novel framework that mimics a human's tactile-guided control strategy. In this framework, we initially collect multi-modal tactile datasets from human demonstrators, incorporating human tactile-guided control strategies for task completion. The subsequent step involves instructing robots through imitation learning using multi-modal sensor data and retargeted human motions. To further mitigate the embodiment gap between humans and robots, we employ online residual reinforcement learning on the physical robot. Through comprehensive experiments, we validate the safety of MimicTouch in transferring a latent policy learned through imitation learning from human to robot. This ongoing work will pave the way for a broader spectrum of tactile-guided robotic applications.

1 Introduction

In the field of robotics and automation, executing intricate tasks like insertion is notably challenging due to dynamic interactions among objects. As a result, a minor execution error can lead to significant task failure, emphasizing the need for adaptive insertion mechanisms with real-time feedback. Many methods, including NVIDIA's "IndustReal" [1] system, have predominantly relied on vision-based solutions [2, 3]. NVIDIA's approach achieves success rates of up to 99.2% in transferring assembly tasks learned in simulations to real-world applications with their customized simulator "Factory" [4]. However, these methods may be limited in environments with compromised visual feedback due to occlusions or varying lighting conditions.

In contrast, humans exhibit innate fine-grained manipulation skills through tactile sensing, allowing for successful insertions by solely using tactile sensations to assess alignment, pose, and force, even without visual input [5]. Motivated by human capabilities, recent studies have explored tactile feedback for complex contact-rich tasks. One combines various sensory modalities in learning from demonstrations [6, 7, 8], and the other employs reinforcement learning (RL) to model insertion tasks [9, 10, 11, 12]. While these studies recognize the significance of tactile feedback, they predominantly rely on robot teleoperation [8, 13, 14] data guided by human visual feedback or pure RL method, which might not utilize human's tactile-guided control strategy.

Learning from human demonstrations has been a long-standing research topic. One particular challenge comes from the embodiment gap, which refers to the morphology disparity between humans and robots. There are two primary methodologies have been developed in utilizing human visual feedback. The first, which involves hybrid datasets, combines human data and teleoperation data to provide a holistic training set [15, 16]. The second is reinforcement learning (RL) fine-tuning [17, 18, 19, 20], which refines pre-trained models through online RL. However, the human

NeurIPS 2023 Workshop on Touch Processing: a new Sensing Modality for AI .

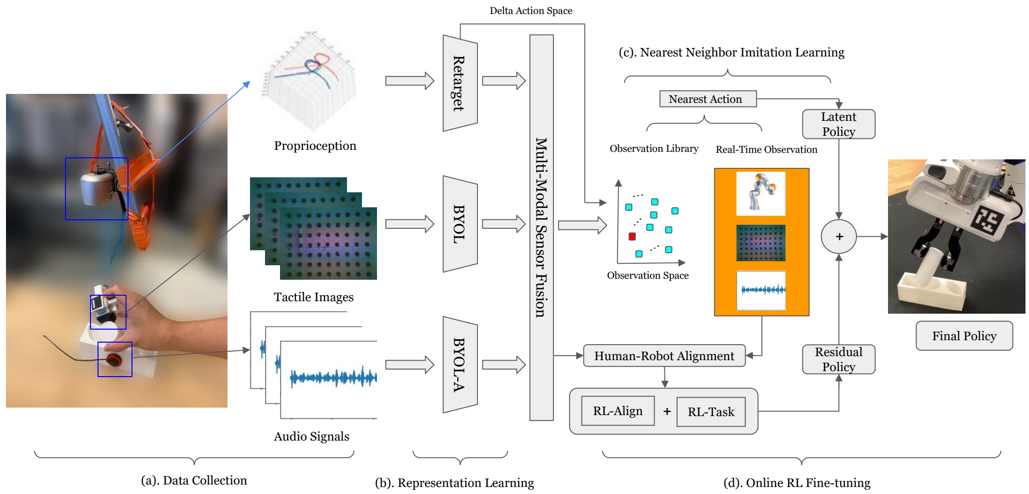

Figure 1: Illustration of our MimicTouch Framework. In part (a), the multi-modal (tactile + audio) tactile feedback is collected during human demonstrations. In part (b), their representation will be learned. In part (c), a latent policy will be learned by Imitation Learning. In part (d), the latent policy will be refined through online residual reinforcement learning on a physical robot.

tactile modality remains relatively uncharted. Furthermore, learning a human's tactile-guided control strategy presents distinct challenges, such as the pronounced embodiment gap in the tactile domain. Given the potential benefits of utilizing human's control strategy guided by tactile feedback, there is a great motivation to solve those challenges and let robots harness human's tactile-guided control strategy.

In response, we introduce "MimicTouch" (in Fig. 1), a novel framework that capitalizes on multi-modal tactile feedback derived from human demonstrations to learn a human's tactile-guided control strategy. The key components of the MimicTouch framework include: 1). A human-centric data collection system, designed to collect multi-modal (tactile + audio) tactile data to learn human's tactile-guided control strategy, 2). A representation learning model that can extract task-specific features and improve the following imitation learning efficiency, 3). A non-parametric imitation learning to learn a latent policy from human demonstrations, 4). A verification tool to evaluate the latent policy in the simulator. 5). An online residual reinforcement learning method learns a residual policy for fine-tuning the latent policy on a physical robot. To evaluate our framework, the ablation study demonstrates enhanced learning efficiency by integrating multi-modal tactile feedback. Also, by employing latent policy evaluation, we show deployable policy transfer to the physical robot. In future work, we will fine-tune our latent policy using online RL and conduct experiments on the physical robot.

2 Methodology

MimicTouch operates through three distinct phases: 1). Representation learning from task-specific human interactions (see Appendix. C), 2). Non-parametric imitation learning with demonstrations from human's tactile-guided control strategy (see Appendix. D), and 3). Online residual reinforcement learning refinement (see Appendix. E). Initially, we use our human data collection system (see Appendix. B) to gather a specialized multi-modal tactile dataset during insertion tasks. Then, we apply two self-supervised learning models to this dataset, producing optimized representation encoders for both tactile and audio data. During the next phase, we leverage imitation learning with multi-modal tactile feedback sourced from our data collection system, aiming to derive a latent policy. Further, we evaluate the safety of this latent policy in a simulated environment (see Sec. 2). The final phase involves the fine-tuning of this latent policy through online residual reinforcement learning. The diagram of this framework is shown in Fig. 1

Data Collection System We design a human-centric data collection system (part (a) in Fig. 1), which simultaneously collects the pose of human fingertips, tactile images, and audio signals when humans use their tactile feedback to complete the insertion tasks. Our advanced data collection system is equipped with specialized hardware for accurate data capture (Shown in Fig. 1). Details of the Hardware Settings and the Fingertips Pose Tracking System are shown in Appendix. B.

Pre-training Tactile Representation To extract low-dimensional useful representations from the demonstration data, we employ SSL, which tries to learn a low-dimensional representation from high-dimensional observations. Specifically, we employ the Bootstrap Your Own Latent (BYOL) [21] for tactile images and BYOL for audio (BYOL-A) for audio data [22], which have demonstrated enhanced performance in computer vision [21], audio representation [22], and robotics [8, 13, 23] domains. Details about BYOL and BYOL-A are shown in Appendix. C.

Leveraging BYOL and BYOL-A, we generate two 1×2048 representation vectors for each tactile image and audio segment. Details are shown in Appendix. C

Learning Latent Policy from Human Demonstration To learn a latent policy from human demonstration, we propose to use the non-parametric imitation learning algorithm to ensure the safety of transferring.

To utilize data from human demonstrations in physical robots, we need to integrate data processing. It consists of three steps: 1). Using data alignment to ensure the tactile images, audio segments, and pose of human fingertips are aligned in the same time stamps. 2). Retargeting the pose of human fingertips into the pose of the robot end-effector. 3). Post-processing the retargeted data for smooth execution. Details are shown in Appendix. D.

Our action space is defined by the $6D$ delta pose transformation of the robot end-effector, encompassing a delta position and a delta Euler angle. Given the inherent challenges of ensuring safety during the transition from human to robot, coupled with the complexity of high-dimensional action and observation spaces, traditional parametric methods often face challenges. These challenges arise from issues such as covariance shifts and the intricacies of learning effective policies in scenarios with limited data. To address these challenges, we employ the nearest neighbors-based imitation learning algorithm (VINN) [24], capitalizing on our collected demonstrations. Details are shown in Appendix. D.

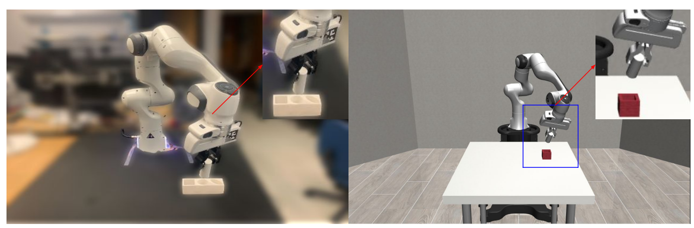

Figure 2: This figure shows our insertion task in both the physical robot environment and simulated environment, which are the same table-top settings containing a GelSight Mini fixture, a cylinder object, and an insertion base.

Evaluation in Simulation Since the current simulator is not able to generate high-quality tactile images and audio data, we need to do the RL fine-tuning directly on a physical robot. Hence, it is important to make sure that the latent policy is deployable to the physical robot environment. To validate the safe transferring of the learned latent policies on the physical robot, we conduct evaluations within a simulated setting. We employ the Robosuite simulation framework [25], underpinned by the Mujoco Physics Engine [26], to simulate the real-world insertion task. The setup is shown in Fig. 2. Details are shown in Appendix. F.

Online Reinforcement Learning Fine-tuning Although we have learned a latent policy from human demonstration, this latent policy might not guarantee task success when deployed on the physical robot. This is partially due to morphology differences between the human and robot end-effector, as well as the inaccurate tracking of the AruCo Marker. Therefore, we aim to further explore online reinforcement learning to enhance the latent policy. However, we cannot directly fine-tune the latent policy because it is learned through a non-parametric imitation learning manner. As a result, we propose to learn a residual policy to solve the embodiment gap issue through RL interactions (part (d) of Fig. 1). While the specifics of the implementation are still under consideration, we outline three potential methods that we believe hold promise in Appendix. E.

3 Experiments

In this section, we present the experiments with our MimicTouch framework.

Descriptions of Manipulation Tasks We evaluate our model in two real-world insertion tasks that require precise tactile feedback. The first one is the Long-horizon Insertion Task. Starting from an initial position, the robot's objective is to navigate to the top of the insertion base and complete the

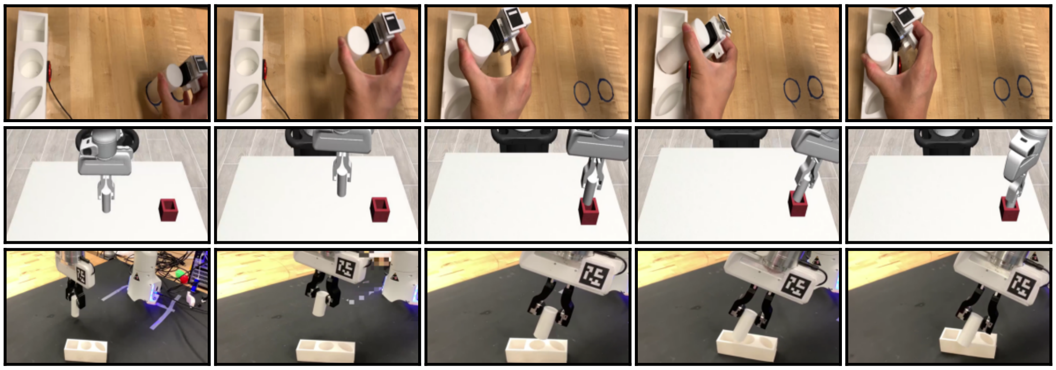

Figure 3: This figure shows our data collection process (first row), the latent policy shown in the simulated environment (second row), and the latent policy shown in the physical robot (third row).

insertion task. One example is shown in Fig. 3. The second one is the Dense Packing. To accomplish this task, the robot is required to initially relocate the object to the available space within the box with some obstacles, and ultimately execute the insertion to the box accurately. One example is shown in Fig. 4. Details of the setups on both tasks are shown in Appendix. G.

Latent Policy Evaluation In this section, we evaluate the latent policy learned from human demonstration. Using the mean square error loss (MSE Loss, defined in Appendix. H), we can evaluate the difference between the generated action sequences and the ground truth action sequences. The results are shown in Table. 1, which shows that our model can generate a latent policy for safe execution. Implementation details and analysis are shown in Appendix. H. The example visualized in the simulation is shown in Fig. 3.

Ablation Study: Do Multi-Modal Tactile Feedback Improve the Performance? We evaluate the performance of our multi-modal tactile feedback with ablation study. The results are shown in Table. 2, which shows that our multi-modal tactile feedback outperforms single modality. Details are shown in Appendix. I.

Physical Robot Experiment In this section, we want to evaluate our approach to the physical robot. The Robot Setup and Calibration are shown in Appendix. J. The setup is shown in Fig. 2.

Firstly, we execute the latent policy on the physical robot. An example is shown in Fig. 3. Currently, we only execute the rollout actions generated from the latent policy in the emulated physical robot environment (shown in Appendix. H) on the physical robot. In the future, we will collect real-time tactile images and audio signals as the input of the latent policy on the physical robot.

According to the example shown in Fig. 3, we can find that it is difficult to complete the task on the physical robot using the latent policy. This is partially due to morphology differences between the human and robot end-effector, as well as the inaccurate tracking of the AruCo Marker. Hence, we need to introduce online fine-tuning to mitigate this issue.

Currently, we are conducting online RL fine-tuning and evaluating our framework with some baseline models (See Appendix. J) and aim to show more promising results by the time of this workshop.

4 Conclusion & Future Work

At present, this progressive project has achieved substantial advancements in the secure transference of human's tactile-guided control strategies to physical robots, with evaluations affirming the efficacy of our proposed multi-modal tactile feedback in enhancing model performance. However, since this is an ongoing project, we acknowledge the limitations such as the latent policy directly generated from human demonstration might not achieve the task success on the physical robot. The next phase of this project will focus on integrating online residual RL to overcome the identified limitations, and we intend to evaluate the overall performance of our proposed approach with other baseline methods. Additionally, we plan to conduct user studies to collect a diverse range of data from various individuals. Built upon our framework, people could explore other potential applications in learning a tactile-guided control strategy for different contact-rich challenging scenes without visual feedback.

References

[1] Bingjie Tang, Michael A. Lin, Iretiayo Akinola, Ankur Handa, Gaurav S. Sukhatme, Fabio Ramos, Dieter Fox, and Yashraj Narang. Industreal: Transferring contact-rich assembly tasks from simulation to reality, 2023.

[2] Gerrit Schoettler, Ashvin Nair, Jianlan Luo, Shikhar Bahl, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. Deep reinforcement learning for industrial insertion tasks with visual inputs and natural rewards, 2019.

[3] Gerrit Schoettler, Ashvin Nair, Juan Aparicio Ojea, Sergey Levine, and Eugen Solowjow. Meta-reinforcement learning for robotic industrial insertion tasks, 2020.

[4] Yashraj Narang, Kier Storey, Iretiayo Akinola, Miles Macklin, Philipp Reist, Lukasz Wawrzyniak, Yunrong Guo, Adam Moravanszky, Gavriel State, Michelle Lu, Ankur Handa, and Dieter Fox. Factory: Fast contact for robotic assembly, 2022.

[5] Gábor Lengyel, Goda Žalalytė, Alexandros Pantelides, James N Ingram, József Fiser, Máté Lengyel, and Daniel M Wolpert. Unimodal statistical learning produces multimodal object-like representations. *eLife*, 8:e43942, may 2019.

[6] Hao Li, Yizhi Zhang, Junzhe Zhu, Shaoxiong Wang, Michelle A. Lee, Huazhe Xu, Edward Adelson, Li Fei-Fei, Ruohan Gao, and Jiajun Wu. See, hear, and feel: Smart sensory fusion for robotic manipulation. In *CoRL*, 2022.

[7] Michelle A Lee, Yuke Zhu, Krishnan Srinivasan, Parth Shah, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks. In *2019 IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[8] Irmak Guzey, Ben Evans, Soumith Chintala, and Lerrel Pinto. Dexterity from touch: Self-supervised pre-training of tactile representations with robotic play, 2023.

[9] Siyuan Dong, Devesh K. Jha, Diego Romeres, Sangwoon Kim, Daniel Nikovski, and Alberto Rodriguez. Tactile-rl for insertion: Generalization to objects of unknown geometry, 2021.

[10] S. Kim and A. Rodriguez. Active extrinsic contact sensing: Application to general peg-in-hole insertion. In *ICRA*, 2022.

[11] Zhao-Heng Yin, Binghao Huang, Yuzhe Qin, Qifeng Chen, and Xiaolong Wang. Rotating without seeing: Towards in-hand dexterity through touch, 2023.

[12] Kaimeng Wang, Yu Zhao, and Ichiro Sakuma. Learning robotic insertion tasks from human demonstration. *IEEE Robotics and Automation Letters*, 8(9):5815–5822, 2023.

[13] Sridhar Pandian Arunachalam, Irmak Güzey, Soumith Chintala, and Lerrel Pinto. Holo-dex: Teaching dexterity with immersive mixed reality, 2022.

[14] Ajay Mandlekar, Yuke Zhu, Animesh Garg, Jonathan Booher, Max Spero, Albert Tung, Julian Gao, John Emmons, Anchit Gupta, Emre Orbay, Silvio Savarese, and Li Fei-Fei. Roboturk: A crowdsourcing platform for robotic skill learning through imitation, 2018.

[15] Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play, 2023.

[16] Kenneth Shaw, Shikhar Bahl, and Deepak Pathak. Videodex: Learning dexterity from internet videos, 2022.

[17] Shikhar Bahl, Abhinav Gupta, and Deepak Pathak. Human-to-robot imitation in the wild, 2022.

[18] Aditya Kannan, Kenneth Shaw, Shikhar Bahl, Pragna Mannam, and Deepak Pathak. DEFT: Dexterous fine-tuning for real-world hand policies. In *7th Annual Conference on Robot Learning*, 2023.

[19] Siddhant Haldar, Jyothish Pari, Anant Rai, and Lerrel Pinto. Teach a robot to fish: Versatile imitation from one minute of demonstrations, 2023.

[20] Siddhant Haldar, Vaibhav Mathur, Denis Yarats, and Lerrel Pinto. Watch and match: Supercharging imitation with regularized optimal transport, 2023.

[21] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning, 2020.

[22] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Byol for audio: Self-supervised learning for general-purpose audio representation, 2021.

[23] Abitha Thankaraj and Lerrel Pinto. That sounds right: Auditory self-supervision for dynamic robot manipulation, 2022.

[24] Jyothish Pari, Nur Muhammad Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The surprising effectiveness of representation learning for visual imitation, 2021.

[25] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning, 2022.

[26] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033, 2012.

[27] Wenzhen Yuan, Siyuan Dong, and Edward H. Adelson. Gelsight: High-resolution robot tactile sensors for estimating geometry and force. *Sensors*, 17(12), 2017.

[28] Elliott Donlon, Siyuan Dong, Melody Liu, Jianhua Li, Edward Adelson, and Alberto Rodriguez. Gelslim: A high-resolution, compact, robust, and calibrated tactile-sensing finger, 2018.

[29] Daolin Ma, Elliott Donlon, Siyuan Dong, and Alberto Rodriguez. Dense tactile force distribution estimation using gelslim and inverse fem, 2019.

[30] Siyuan Dong and Alberto Rodriguez. Tactile-based insertion for dense box-packing, 2019.

[31] Sudharshan Suresh, Zilin Si, Joshua G. Mangelson, Wenzhen Yuan, and Michael Kaess. Shapemap 3-d: Efficient shape mapping through dense touch and vision, 2022.

[32] Edward J. Smith, Roberto Calandra, Adriana Romero, Georgia Gkioxari, David Meger, Jitendra Malik, and Michal Drozdzal. 3d shape reconstruction from vision and touch, 2020.

[33] Edward J. Smith, David Meger, Luis Pineda, Roberto Calandra, Jitendra Malik, Adriana Romero, and Michal Drozdzal. Active 3d shape reconstruction from vision and touch, 2021.

[34] Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H. Adelson, and Sergey Levine. More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and Automation Letters*, 3(4):3300–3307, oct 2018.

[35] Lerrel Pinto, Dhiraj Gandhi, Yuanfeng Han, Yong-Lae Park, and Abhinav Gupta. The curious robot: Learning visual representations via physical interactions, 2016.

[36] Roberto Calandra, Andrew Owens, Manu Upadhyaya, Wenzhen Yuan, Justin Lin, Edward H. Adelson, and Sergey Levine. The feeling of success: Does touch sensing help predict grasp outcomes?, 2017.

[37] Yunhai Han, Kelin Yu, Rahul Batra, Nathan Boyd, Chaitanya Mehta, Tuo Zhao, Yu She, Seth Hutchinson, and Ye Zhao. Learning generalizable vision-tactile robotic grasping strategy for deformable objects via transformer, 2023.

[38] Jivko Sinapov, Mark Wiemer, and Alexander Stoytchev. Interactive learning of the acoustic properties of household objects. In *2009 IEEE International Conference on Robotics and Automation*, pages 2518–2524, 2009.

[39] Joshua Christie and Navinda Kottege. Acoustics based terrain classification for legged robots. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3596–3603, 2016.

[40] Carolyn Matl, Yashraj Narang, Dieter Fox, Ruzena Bajcsy, and Fabio Ramos. Stressd: Sim-to-real from sound for stochastic dynamics, 2020.

[41] Dhiraj Gandhi, Abhinav Gupta, and Lerrel Pinto. Swoosh! rattle! thump! – actions that sound, 2020.

[42] Amrita Sawhney, Steven Lee, Kevin Zhang, Manuela Veloso, and Oliver Kroemer. Playing with food: Learning food item representations through interactive exploration, 2021.

[43] Yuzhe Qin, Yueh-Hua Wu, Shaowei Liu, Hanwen Jiang, Ruihan Yang, Yang Fu, and Xiaolong Wang. Dexmv: Imitation learning for dexterous manipulation from human videos, 2022.

[44] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning, 2016.

[45] Mandy Xie, Ankur Handa, Stephen Tyree, Dieter Fox, Harish Ravichandar, Nathan D. Ratliff, and Karl Van Wyk. Neural geometric fabrics: Efficiently learning high-dimensional policies from demonstration. In Karen Liu, Dana Kulic, and Jeff Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 1355–1367. PMLR, 14–18 Dec 2023.

[46] Yunhai Han, Mandy Xie, Ye Zhao, and Harish Ravichandar. On the utility of koopman operator theory in learning dexterous manipulation skills. *arXiv preprint arXiv:2303.13446*, 2023.

[47] Yanlong Huang, Leonel Rozo, João Silvério, and Darwin G. Caldwell. Non-parametric imitation learning of robot motor skills. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5266–5272, 2019.

[48] Maarten Vaandrager, Robert Babuška, Lucian Buşoniu, and Gabriel A.D. Lopes. Imitation learning with non-parametric regression. In *Proceedings of 2012 IEEE International Conference on Automation, Quality and Testing, Robotics*, pages 91–96, 2012.

[49] Gelsight mini. `https://www.gelsight.com/gelsightmini/`.

[50] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognit.*, 47:2280–2292, 2014.

[51] Manish Sinha Kanishke Gamagedara, Jean Gressmann. aruco-markers. `https://github.com/fdcl-gwu/aruco-markers.git`, 2018.

[52] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

Appendices

A Related Work

In this section, we contextualize our contributions within relevant sub-fields.

A.1 Multi-Modal Tactile Sensing

Vision-based tactile sensing is integral to robotic manipulation, with sensors equipped with internal cameras offering high-resolution data on local contact geometry and frictional properties [27, 28, 29, 30]. Through the application of tactile sensing, several learning-based methodologies have been developed for a range of manipulation tasks.

Various studies have incorporated imitation learning algorithms with tactile sensing in multi-modal perception to accomplish intricate tasks such as insertion, dense packing, pouring, and other dexterous manipulation tasks [6, 7, 8]. On the other hand, some researchers have focused on employing tactile feedback solely as the reward function of reinforcement learning, successfully completing tasks like insertion, in-hand rotation, and multi-material object cutting without visual feedback [9, 10, 11]. Out of learning-based methodologies, tactile sensing is also widely used in shape reconstruction [31, 32, 33], grasping [34, 35, 36, 37], and dense packing [30]. Nonetheless, the vision-based tactile sensor may not fully represent the features of tactile sensing.

Audio-based tactile sensors, e.g. contact microphones, have been demonstrated to be another effective modality in robotics, such as robot manipulation [23], classifying object instances [38, 39], modeling dynamics [40, 41], and learning food embedding [42]. Such data can emulate nerve endings within human skin that detect vibrations during tactile interactions, enhancing human's perception of surface roughness. Furthermore, it facilitates the detection of collisions and slippages between objects [41]. Incorporating this into multi-modal tactile sensing can yield tactile feedback more akin to human sensations, which is helpful for learning a human's tactile-guided control strategy.

A.2 Learning from Human Demonstration

Leveraging human demonstration offers an avenue to leverage human's tactile-guided control strategy. The domain of learning from human demonstrations, especially from videos [43], has been extensively researched. Given the ease of collecting human demonstrations, the learning process is considerably less time-intensive and more diverse compared to learning from teleoperation data and pure reinforcement learning.

Two prominent methods include hybrid learning from both human and robot teleoperation [15, 16] and model fine-tuning using reinforcement learning [12, 17, 18, 19, 20]. Both methods effectively address the embodiment gap challenges when transitioning from human to robot. In this study, we focus on leveraging genuine human's tactile-guided control strategy, eschewing robot teleoperation data, and opting for reinforcement learning-based fine-tuning.

A.3 Imitation Learning

For the purpose of our study, it is imperative to employ an imitation learning algorithm to derive a latent policy from human demonstrations. Imitation learning can be bifurcated into two categories: parametric [44, 45, 46] and non-parametric imitation learning [24, 47, 48]. Parametric imitation learning algorithms, which rely on computing a multitude of related parameters, offer superior generalizability. However, they are also prone to a higher covariance shift, making them potentially hazardous when applied directly to out-of-domain executions. In contrast, non-parametric imitation learning algorithms derive insights solely from the observations and trajectories in the dataset, devoid of any additional parameters. While they may be short of generalizability, they offer a safer alternative to their parametric counterparts. Considering our objective of safely learning human's tactile-guided strategy, we have opted for the non-parametric VINN algorithm [24].

B Data Collection System

Hardware Our data collection system is equipped with specialized hardware for data capture (shown in Fig. 1). The data encompasses three elements: human fingertip pose, which will be retargeted to the pose of the robot end-effector; tactile images, which indicate object-sensor contact; and audio signals, which mimic the vibrations detected by deep skin nerve endings. We use the RealSense camera for fingertip pose tracking, offering a 60 Hz stream at a 320x240-pixel resolution. The GelSight Mini [49], a compact vision-based tactile sensor, facilitates tactile imaging with a 30 Hz stream at 400x300 pixels. This sensor is conveniently mounted onto human fingertips using a custom fixture (refer to Fig. 1). Audio data is captured using the HOYUJI TD-11 piezo-electric contact microphone with a 44.1kHz sampling rate. Due to the fact that the robot's inherent noise differs from that of humans, the microphone is placed at the base of our insertion task (shown in Fig. 1) instead of the end-effector or human fingertips, ensuring consistent audio data collection. For our insertion tasks in Appendix. G, the vibration from collisions and slippery will be only helpful to robot policy during the final insertion process when physical contact happens but not during the previous contact-free process. Hence, installing it on the base represents a part of multi-modal tactile sensing and helps learn human's tactile-guided control strategy.

Fingertips Pose Tracking To track the human fingertip motions, we put an AruCo marker [50] on our custom fixture to track human fingertip movements. Positioned in a tabletop view, the RealSense camera captures the AruCo marker's pose. Using an open-source library [51], the marker's 6D pose was tracked at 60Hz. These human fingertip poses are subsequently mapped to the robot end-effector's pose, serving as vital input for the imitation learning process after retargeting and post-processing (shown in Appendix. D).

C Pre-training Tactile Representation

Data Collection For self-supervised learning (SSL) training, we collect task-specific tactile-audio data obtained from human's tactile-guided insertion tasks shown in the Appendix. G. Our dataset encompasses success, failed, and sub-optimal instances. Each task yields tactile images at 30 Hz and audio data sampled at 44.1kHz, segmented at 2Hz. In total, we collected 9,157 tactile images and 1,000 audio segments for SSL training.

Representation Learning To extract low-dimensional useful representations from the play data, we employ SSL, which tries to learn a low-dimensional representation from high-dimensional observations. Specifically, we employ the Bootstrap Your Own Latent (BYOL) [21] for tactile images and BYOL for audio (BYOL-A) for audio data [22], which have demonstrated enhanced performance in computer vision [21], audio representation [22], and robotics [8, 13, 23] domains.

BYOL [21] generates two augmented views, $\mathrm{v} \triangleq \mathrm{t}(\mathrm{x})$ and $\mathrm{v}' \triangleq \mathrm{t}'(\mathrm{x})$, from a given x by applying image augmentations $\mathrm{t} \sim \mathcal{T}$ and $\mathrm{t}' \sim \mathcal{T}'$ respectively, where $\mathcal{T}$ and $\mathcal{T}'$ represent distinct augmentation distributions. The architecture of BYOL comprises a primary encoder f_θ and a target encoder f_ξ, where the latter being an exponential moving average of the former. Given the augmented views v and v', they are processed to yield representations y and y'. These representations are subsequently transformed by projectors g_θ and g_ξ to produce higher-dimensional vectors z and z'. The primary encoder and its associated projector are designed to predict the output from the target projector, resulting in $\mathrm{q}_\theta(\mathrm{z}_\theta)$ and $\mathrm{sg}(\mathrm{z}'_\xi)$. The model's output consists of l_2-normalized versions of these predictions, which are trained using a similarity loss function. Post-training, the encoder f_θ is utilized for feature extraction from observations.

To utilize BYOL in tactile images, we scale the tactile image up to 256x256 to work with standard image encoders. We use the ResNet [52] architecture, also starting with pre-trained weights. Unlike SSL techniques in visual images, we only apply the Gaussian blur and small center-resized crop augmentations, since other augmentations such as color jitter and grayscale would violate the assumption that augmentations do not change the tactile signal significantly. For each input, the trained model will generate a 1×2048 representation vector.

BYOL-A [22] is an extended version of BYOL to audio representation learning, processing log-scaled mel-spectrograms through a specialized augmentation module. To utilize BYOL-A in our audio data, we down-sampled signals from 44.1kHz to 16kHz, with a window size of 64 ms, a hop size of 10 ms, and mel-spaced frequency bins F = 64 in the range 60–7,800 Hz. Then, the Pre-Normalization step stabilizes the input audio for subsequent augmentations. Once normalized, the Mixup step introduces contrasts in the audio's background, defined by the log-mixup-exp formula:

$$\tilde{\mathrm{x}}_{\mathrm{i}} = \log((1-\lambda)\exp(\mathrm{x}_{\mathrm{i}}) + \lambda \exp(\mathrm{x}_{\mathrm{k}}))$$

where x_{k} is a mixing counterpart and λ is a ratio from a uniform distribution. The next one is the RRC block, an augmentation technique, that captures content details and emulates pitch shifts and time stretches. For each input, the trained model will generate a 1×2048 representation vector.

D Learning Latent Policy from Human Demonstration

D.1 Data Processing

To utilize data from the human demonstrations in learning a robot policy, we need to integrate data processing.

Data Alignment To ensure synchronization across our sensors, we address the disparate sampling rates of the RealSense, tactile sensor, and audio sensor, which are 60Hz, 30Hz, and 2Hz, respectively. Given the continuous nature of the insertion tasks, a 2Hz sampling rate is insufficient. We downsample the pose and tactile image data to 5Hz. For audio, rather than collecting discrete 0.5s segments, we only capture extended audio data for every 0.2s, resulting in a 0.3s overlap between every two neighboring audio segments. Finally, we obtain one human fingertip pose, one tactile image, and one 0.5s audio segment at 5Hz.

Pose retargeting The 6D human fingertip poses that are extracted from the AruCo marker encompass 3D positions along with rotation vectors. Then, we transform these rotation vectors into Euler angles. The transformed poses function as desired robot end-effector poses.

Post Processing Given the inherent noise and occasional outliers in the poses obtained from the RealSense and AruCo markers, it is imperative to implement post-processing techniques to ensure the quality and smoothness of the trajectories. For each pose sequence, outliers are detected by sorting the values of each delta transformation. The Interquartile Range (IQR) method is employed to establish the upper and lower bounds, which are then used to identify outliers. The IQR is defined as: IQR $= Q_3 - Q_1$ where Q_3 and Q_1 are the third and first quartiles, respectively. Outliers are replaced using a median filter with a window size of 3. To enhance the temporal consistency of the estimated hand and object pose, a digital low-pass filter is applied to eliminate high-frequency noise. Specifically, the filter has a sampling frequency of 5Hz and a cutoff frequency of 2Hz. The low-pass filter can be represented as:$H(f) = \frac{1}{1+(\frac{f}{f_c})^2}$ where f is the sampling frequency and f_c is the cutoff frequency.

D.2 Non-Parametric Imitation Learning

The observations and actions in the collected demonstrations are shown as $(\mathrm{o}_i^T, \mathrm{o}_i^A, \mathrm{o}_i^e, \mathrm{a}_i)$ at the i-th step, where the o_i^T is the tactile image, the o_i^A is the audio segment, the o_i^e is the robot end-effector pose, and a_i is the action. Then, we extract tactile and audio features $(\mathrm{y}_i^T, \mathrm{y}_i^A)$ from the observations using our pre-trained encoders (shown in Appendix. C). These features and the corresponding robot end-effector pose serve as the input $(\mathrm{y}_i^T, \mathrm{y}_i^A, \mathrm{o}_i^e)$ to our model for generating the next action a_{i+1}. Given the varying scales of these inputs, we apply normalization such that the maximum distance for each input is unity in the dataset. During testing and experiment, for a given real-time observation $\hat{\mathrm{o}}_t$, we compute the features $(\hat{\mathrm{y}}_t^T, \hat{\mathrm{y}}_t^A, \hat{\mathrm{o}}_t^e)$. Then, we use these features to identify the data point in the dataset with the minimal aggregate distance, and subsequently execute the associated action.

E Online Reinforcement Learning Fine-tuning

E.1 Residual RL via Expert-Aligned Rewards

In recent research endeavors, a promising approach involves fine-tuning the latent policy through real-world interactions, aiming to align it with an expert policy [12, 19, 20]. These works employ optimal transport-based reinforcement learning to fine-tune the latent policy derived from expert data with expert-aligned rewards. Experimental results suggest that this technique can efficiently generate a robust policy, adaptable to analogous tasks across diverse environments. This is achieved with minimal training duration, even when the latent policy is derived from a small dataset.

Such algorithms demonstrate the capacity to efficiently learn policies that are generalizable across similar tasks in diverse environments. A salient feature of this approach is its emphasis on aligning the robotic policy with that of experts, ensuring the assimilation of safety-centric policies from these experts. This safety alignment is important, especially when dealing with out-of-domain datasets. However, the efficacy of these methods has been primarily validated using the in-domain training data, which only encompasses configuration gaps from visual imagery. Their applicability to out-of-domain datasets, especially those with embodiment gaps remains unexplored. Due to the embodiment gap, the expert policies, which are retargeted from human demonstrations, often prove incompatible with direct robotic applications. Consequently, the endeavor to align robotic policies with potentially non-transferable datasets may not effectively address the embodiment gap issue.

E.2 Residual RL via Task-Based Rewards

The task-based rewards represent a prominent methodology for refining the latent policy and facilitating the completion of insertion tasks. In related studies [9, 10], the authors introduce a tactile-based feedback insertion policy, utilizing reinforcement learning (RL) to model the insertion process as an episodic policy alternating between insertion attempts and pose corrections. With experiments evaluated on the physical robot, those papers show superior performance with RL using task-based rewards computed from tactile images.

Besides, from the paper [12], we see numerous constraints were introduced to ensure training safety. In such intricate tasks in Appendix. G, to encourage the RL exploration while incorporating other safety constraints presents a great challenge. Furthermore, task-based rewards are ideally sparse in nature. In our task, the robots need to constantly adjust their pose, which is a very continuous motion. As a result, using sparse rewards might exacerbate training complexities and reduce overall training efficiency.

E.3 Residual RL via Hybrid Expert-Aligned Rewards and Task-based Rewards

The limitations of either expert-aligned rewards or task-based rewards are shown in Appendix. E, the hybrid method has been driven by several studies [17, 18]. In those studies, the authors employ both expert-aligned rewards and task-based rewards to refine the policy generated from human demonstration. Such a hybrid approach enhances the strengths of both methodologies, facilitating the extraction of task-based rewards to aid in task completion while concurrently upholding training safety standards. This potential approach is shown in part (d) of Fig. 1.

The hybrid strategy has been proven that it can utilize both advantages of different rewards with human visual feedback. It not only can ensure safe training and fast fine-tuning but also can solve the embodied gap. We hope that this method can also play a good role in fine-tuning the human demonstration with tactile feedback.

F Simulation Setup

As depicted in Fig. 2, a tabletop manipulation setup is established in the simulation to replicate the insertion task. The end-effector's fingers are replaced with our custom fixture designed for the GelSight Mini. The object is fixed in proximity to the end-effector to prevent slippage during interactions. Given Mujoco's constraint of supporting only convex objects, we construct a base with an insertion hole using four small cuboids. This setup allows us to execute the latent policy seamlessly. Details about latent policy and safety evaluations are shown in Appendix. H.

G Decriptions of Manipulation Tasks

To evaluate our model in two real-world insertion tasks that require precise tactile feedback.

Long-horizon Insertion Task We 3D print three objects: a cylinder, a cuboid, and an elliptical cylinder with their hole base as shown in Fig. 2. We set up the task environments for both data collection in Appendix. B and physical robot experiments (an example shown in Fig. 3). Starting from an initial position, the robot's objective is to move to the top of the insertion base and complete the insertion task. This task is notably more challenging than standard insertion tasks. It requires the robot to emulate a blindfolded human who needs to locate the insertion point and continuously adjust the object's posture until finishing insertion. In contrast, traditional insertion tasks typically proceed without considering pose variations and consistently keep the object perpendicular to the table. For the data collection, we attempt to blindfold and insert the object with our tactile feedback to collect a dataset to learn a human's tactile-guided control strategy. An example is shown in Fig. 3.

Dense Packing The objective of this task is to accurately insert an object into a constrained space within a nearly saturated box. To accomplish this goal, the robot is required to initially relocate the object from its starting position to a point above the box, subsequently navigate around obstacles, identify the available space within the box, and ultimately execute the insertion accurately. We conduct two variations of dense packing: one employing 3D printed objects, facilitating a more controlled examination of each modality's role in this task, and another utilizing diverse real-world objects. An example is shown in Fig. 4. We will implement this task after evaluating the long-horizon insertion task.

Figure 4: An example for the dense packing task.

H Latent Policy Evaluation

In this section, we evaluate the latent policy learned from the human demonstration. Using the mean square error loss (MSE Loss), we can evaluate the difference between the generated actions and the ground truth actions.

Emulate Physical Robot Environment Using the data collection system (Appendix. B), we collect 20 noisy-free data sequences with the trained encoder (Appendix. C) as the datasets to learn the latent policy. For policy evaluation, we gather another 10 data sequences with random noise, which helps to emulate the physical environments in the real world.

To emulate the physical robot environment, we introduce random noise to those 10 unseen data sequences. The robot state space input undergoes a random position noise within the range $[-3\text{cm}, +3\text{cm}]$ for each axis. Gaussian noise, denoted as $\mathcal{N}(0, \sigma)$, is added to both the tactile image and audio signal. In this notation, $\mathcal{N}(0, \sigma)$ signifies a Gaussian distribution with a mean of 0 and a standard deviation of σ. For tactile images, the noise affects pixel values in the range $[0, 255]$, while for audio data, it impacts signal values in the range $[0, 1]$. Given their distinct ranges, we apply Gaussian noise with standard deviations of $\sigma = 100$ for tactile images and $\sigma = 0.4$ for audio data.

Generate Action Sequences To evaluate the latent policy with those unseen data sequences with noises, we follow these steps: 1). For the first initial pose, we use the collected pose, tactile image, and audio data, all subjected to noise. 2). For subsequent poses, our VINN model is employed to generate a new pose utilizing the pose and sensor data from the preceding state. Then, we add noises on the newly synthesized pose and the multi-modal tactile data derived from the original unseen action sequence corresponding to the current time step. The synthesized pose and tactile data with noises will be used for generating the next pose. 3). After obtaining a trained sequence with the same length as the original unseen data sequence, we finish the generation of action sequences.

MSE Loss For calculating the MSE Loss between two action sequences, we need to normalize the actions' translation vectors and rotation vectors since they have different scales. Then, we calculate the average MSE Loss between each action of these two action sequences. The formula is shown as:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{y}_i - \hat{\mathrm{y}}_i)^2$$

Where: y_i represents the ground truth action, $\hat{\mathrm{y}}_i$ represents the generated action, and n is the number of all action steps.

Evaluate the Latent Policy To assess the efficacy and safety of the latent policy, we evaluate the actions generated by the latent policy with the data sequence in the emulated simulated environment using the MSE Loss (reported in Table. 1) between the generated action sequence and the ground truth action sequence.

We test three modes of generating the actions from the latent policy: 1). Use the recorded observation in the unseen data sequence to generate the next action, 2). Use the rollout robot state and the recorded sensor data to generate the next action, and 3). Use the noisy rollout robot states and noisy recorded sensor data to generate the next action (same method as Appendix. H shown above). From the table. 1, since all of those MSE Loss are similarly small, it shows that the latent policy can generate actions that are similar to the ground truth even with random noise. This indicates that the policy would **not** generate risky behaviors for the subsequent online RL fine-tuning on the physical robot.

States	Noise	MSE Loss
Desired States	No	0.14
Current States	No	0.21
Current States	Yes	0.26

Table 1: Average MSE Loss of 10 action sequences.

Additionally, we visualize the retargeted action sequence and generated action sequence (mode 3) in the Robosuite simulated environment in Fig. 3, which shows that the rollout poses generated using VINN are close to the ground-truth retargeted pose.

I Ablation Strudy: Do Multi-Modal Tactile Feedback Improve the Performance?

In this section, we want to evaluate the performance of our multi-modal tactile embeddings.

Same as the Appendix. H, we evaluate the performance of our multi-modal tactile embeddings with Mean Square Error of action sequence generated from unseen action sequence collected by humans with random noises. We evaluate our model with the following baseline models:

- **MimicTouch w/o T & A:** MimicTouch without tactile or audio embeddings.
- **MimicTouch (T):** MimicTouch incorporating only tactile embeddings.
- **MimicTouch (A)** MimicTouch incorporating only audio embeddings.
- **MimicTouch (T + A, Ours):** MimicTouch incorporating both tactile and audio embeddings.

From Table. 2, we report the MSE Loss between the generated actions and the ground truth actions. Without using both tactile images and audio signals, the MSE loss reaches 0.62, which is much higher than the others. In the case of using multi-modal tactile sensing, our approach outperforms the variants with a single modality.

Models	MSE Loss
MimicTouch w/o T & A	0.62
MimicTouch (T)	0.38
MimicTouch (A)	0.48
MimicTouch (T + A, Ours)	0.26

Table 2: Average MSE Loss of 10 action sequences in the ablation study.

J Physical Robot Evaluation

Robot Setup All experiments are conducted on a Franka Emika Panda Arm. An inverse kinematics solver is used to map the 6-DoF Cartesian space displacement commands into the 7-DoF joint actions. For each task, Cartesian space displacement commands are generated at a policy frequency of 5 Hz, subsequently mapped by the low-level control loop to force and torque inputs in the robot's joint space at 200 Hz.

Calibration Given that our data collection and final experiments occur in disparate scenarios, it's crucial to align our human-centric data collection system with the physical robot system. Initially, we record the distance between the object (starting point) and the base (ending point) within the data collection system and replicate this setup in the robot environment. Following this, six equidistant positions between the starting and ending points are identified within both systems. The object is gripped at these predetermined positions using both hands and the robot's end-effector, allowing us to capture the corresponding poses. In this calibration process, the hand poses, denoted as the "Eye" in the calibration function, are referenced to the camera frame, while the end-effector poses, represented as the "Hand" in the calibration system, are referenced to the robot frame. Conclusively, we employ the calibrateHandEye function from OpenCV, using the six captured poses, to calibrate these two frames (camera frame and robot frame).

Baseline Models We propose to compare our model with the following baselines:

- **Tactile-RL** [9]: A baseline method uses pure RL policy generated from tactile images.
- **Active iSAM** [10]: A method estimates the contact line with tactile images, and uses the RL policy generated from the contact line.
- **MimicTouch w/o RL**: A variant of our model without RL fine-tuning
- **MimicTouch (T)**: A variant of our model with only vision-based tactile feedback.
- **MimicTouch (A)**: A variant of our model with only audio-based tactile feedback.
- **MimicTouch (T+A, Ours)**

Notably, since our framework only uses human demonstration guided by tactile feedback, some other works that use visual feedback and teleoperation such as MULSA [6] are not considered.