

Stick Roller: Precise In-hand Stick Rolling with a Sample-Efficient Tactile Model

Yipai Du¹, Pokuang Zhou², Michael Yu Wang^{3,4}, *Fellow, IEEE*, Wenzhao Lian⁵, Yu She²

Abstract—In-hand manipulation is challenging in robotics due to the intricate contact dynamics and high degrees of control freedom. Precise manipulation with high accuracy often requires tactile perception, which adds further complexity to the system. Despite the challenges in perception and control, the rolling stick problem is an essential and practical motion primitive with many demanding industrial applications. This work aims to learn the high-resolution tactile dynamics of the rolling stick. Specifically, we try manipulating a small stick using the Allegro hand equipped with the Digit vision-based tactile sensor. The learning framework includes an action filtering module, tactile perception module, and learning with uncertainty module, all designed to operate in low data regimes. With only 2.3% amount of data and 5.7% model complexity of previous similar work, our learned contact dynamics model achieves better grasp stability, sub-millimeter precision, and promising zero-shot generalizability across novel objects. The proposed framework demonstrates the potential for precise in-hand manipulation with tactile feedback on real hardware. The project source code is available at: https://github.com/duyipai/Allegro_Digit. A video presentation is available here.

I. INTRODUCTION

In-hand manipulation is a crucial skill for humans to manipulate objects, allowing people to utilize various tools. Robotic in-hand manipulation will play an essential role in extending the deployment of robots to accomplish more complex industrial tasks. The parallel-jaw grippers can utilize extrinsic contact [1] or dynamic manipulation [2], [3] to regrasp an object to change its relative pose in hand, but multiple degrees of freedom in the fingers can enable more flexibility and precision (Fig. 1), especially in confined space. The motion primitive of rolling a stick-like object between two fingers can be used to tighten or loosen the screws or change the pose of the grasped object after the initial grasp. It can significantly improve the flexibility of robotic systems in various settings. For example, the manipulator has to move back and forth to adjust the pose of the in-hand cable [4], but it is unsuitable for confined space industrial manipulation. In that case, the in-hand adjustment becomes the optimal motion to adopt. Although this task seems simple, it is non-trivial to achieve which usually requires accurate tactile



Fig. 1. The hardware setup for this work. Two fingers of the Allegro robotic hand are used to manipulate a piece of stick. The Digit vision-based tactile sensor on the thumb finger reads the contact state and enables manipulation without visual feedback

feedback for high precision grasps [5]. Robotic hands' high degrees of freedom increase their flexibility but also induce challenges in designing control policies. The precision requirement poses difficulty to sim-to-real approaches [6] and makes high-resolution tactile sensing indispensable. Among various types of tactile sensors, vision-based tactile sensors can provide superior sensing resolution at a low design and fabrication cost [7]. Although many prior works have tried to reduce the learning complexity of in-hand manipulation, few have focused on capturing the fine contact dynamics within millimeters, which is crucial for tasks requiring precise contact control within the fingertip.

We hope to rely on high-resolution tactile feedback and a generic robotic hand to solve the task of precisely rolling a stick-like object, which has a potential impact on many industrial applications. The primary contributions of this paper include:

- A method to accurately estimate the grasping strength and pose of stick-like objects in hand from tactile feedback
- A framework to learn from a small amount of data while achieving high precision in learning tactile dynamics
- A real-hardware experimental study to show the proposed method can enable rolling object pose control with submillimeter accuracy

¹ The Hong Kong University of Science and Technology, Hong Kong.

² Purdue University, 610 Purdue Mall, West Lafayette, IN 47907, USA.

³ HKUST Shenzhen-Hong Kong Collaborative Innovation Research Institute, Futian, Shenzhen, China.

⁴ School of Engineering, Great Bay University, Songshan Lake, Dongguan, Guangdong, China.

⁵ University of Chinese Academy of Sciences, China.

This work was supported by the Project of Hetao Shenzhen-Hong Kong Science and Technology Innovation Cooperation Zone, under Grant HZQB-KCZYB-2020083. Partial financial support was received from Google. Corresponding author: Yu She. Email: shey@purdue.edu

The proposed learning framework includes action filtering, model-based tactile signal extraction, and uncertainty-aware dynamics learning. In addition to safely and precisely achieving the rolling stick motion, we hope to explore the potential of learning tactile in-hand manipulation on real hardware, which is inevitable for learning precise contact dynamics.

This paper is structured as follows. Section II reviews related work on vision-based tactile signal processing and in-hand manipulation. In Section III, we introduce the three essential modules of the framework, each specifically designed to facilitate efficient learning. In Section IV, quantitative experiments are conducted to validate the accuracy of the learned contact dynamics model, using both test data and hardware tests. We also extend the experiments to include novel object testing and pose control testing, demonstrating the usefulness of the learned model for manipulating unseen objects. Section V gives the conclusion of the work.

II. RELATED WORKS

Robotic in-hand manipulation has been considered necessary due to its flexibility for various daily tasks. Early studies rely on task-specific hardware and control algorithms to achieve fast and accurate object manipulation [8]. In recent years, stronger and more robust robotic hands have been developed [9], allowing the exploration of more generic human-like tasks. To effectively utilize the high degrees of freedom in the place of complex contact dynamics, reinforcement learning approaches are becoming popular. For example, OpenAI [10] demonstrated for the first time using end-to-end deep-RL based method for controlling finger-gaiting and finger-pivoting, but the training and sim-to-real transfer are difficult to reproduce outside the lab. Therefore, the core question in multi-finger in-hand manipulation has been improving the sample efficiency of learning in high dimensionality [11] to make the methods more scalable and manageable. Chen *et al.* [12] studied the human hand kinematics and dynamics constraints, which inspired novel robotic hand hardware and algorithm design to reduce the action complexity. Sub-optimal controllers using domain knowledge [13] and demonstration through human hand teleoperation [14] were also proven to be effective in improving the sample efficiency in reinforcement learning for in-hand manipulation. Shaw *et al.* [15] extracted basic manipulation primitives from internet videos as visual, action, and physical priors. Many techniques [16]–[18] on sim-to-real transfer were also emerging that allow the learned models in simulation to be deployed on real hardware. A problem with these systems above is that they are designed for palm-scale in-hand manipulation rather than fingertip-scale, and capturing the fine contact dynamics within the fingertip has not yet been achieved by sim-to-real. Hence, they are not suitable for precise manipulation. We learn the physics model in real life to capture the precise dynamics at the fingertips, thus reducing the gap between learning and deployment.

High-resolution tactile sensing is desired as feedback to control the subtle contact within the fingertips. Vision-based tactile sensors using digital cameras to capture the sensing

surface deformation are promising as they have high sensing resolution, easy manufacturing methods, and robustness in harsh settings. It also synergizes tactile signal processing with visual signal processing tools [19]. Various kinds of vision-based tactile sensors [20]–[24] are being developed that satisfy different needs on the tactile signal perception. Among them, the Digit tactile sensor [25] achieved enhanced reliability and compact design that can fit into the Allegro hand fingertip. Lambeta *et al.* [25] adopted an autoencoder structure to extract the contact center and strength for the Digit tactile sensor in a self-supervised manner. They achieved the manipulation of a marble ball within the two fingertips but consumed more than 10 hours of real data. We hope to solve the rolling stick problem, which is more difficult than marble manipulation due to its smaller size and less concentrated weight, in a data-efficient manner to make learning precise contact dynamics more scalable on real hardware.

III. METHOD DESCRIPTION

The proposed method for learning precise tactile dynamics relies on three key components: action filtering, tactile perception, and learning the dynamics model with uncertainty, each specifically designed to enhance learning efficiency. The whole pipeline of the learned model is shown in Fig. 2.

A. Action Filtering

This work aims to focus the effort on the part of the action space that is more conducive to maintaining a stable grasp. To achieve this, we reduced the 8 DoFs (degrees of freedom) two-finger system to 7 DoFs by fixing the base joint of the index finger. This is because simultaneously moving the base joints of the index finger and the thumb is unnecessary for this task. Then, we form the choice of actions by uniformly sampling with a step size of 0.1 radians in the 7 DoFs joint space. The actions were filtered by calculating the forward kinematics of the fingers. Although the forward kinematics of robotic hands are usually not as precise as robotic arms', mainly due to less powerful hardware, it is still a helpful tool for in-hand manipulation to remove the irrelevant robot finger configurations. The filtering was based on a box constraint to make the two fingertips close to each other and an orientation constraint to make the two fingertips roughly facing each other. The upper and lower bounds for these two constraints were naively adjusted by hand. Still, the method effectively reduced the action space as most configurations did not satisfy these two simple constraints (see Section IV-A). This idea can be extended to use learned classifiers (filters) from demonstrations [14] for space reduction in more complex tasks and will be a future work.

B. Tactile Perception

The Digit tactile sensor [25] developed by Meta was embedded into the Allegro robotic hand. The sensor captures RGB images at a resolution of 240x320 and has a pixel size of 0.0487mm per pixel. We considered a strategy that exploits the contact geometry model to achieve efficient

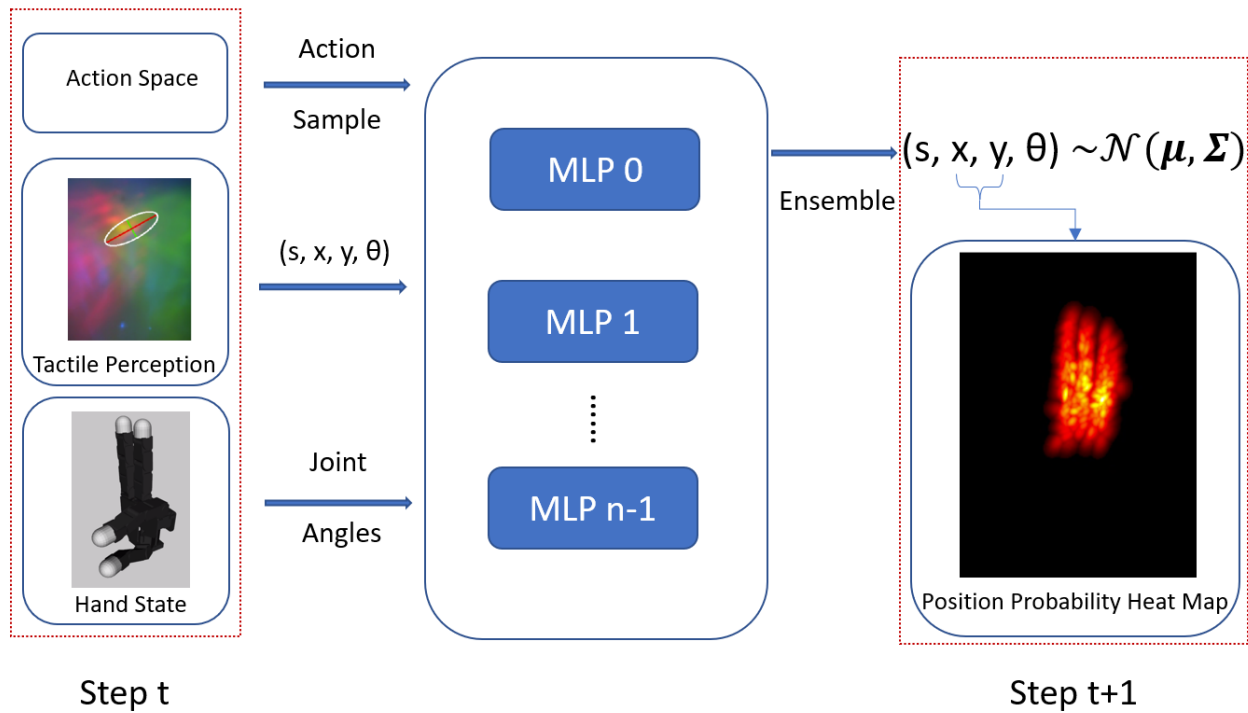


Fig. 2. The pipeline for the tactile dynamics model. The tactile processing outputs (s, x, y, θ) represent the contact strength, contact center, and orientation. The action sample, tactile states, and hand state are fed into multiple multilayer perceptrons to output the future tactile state distribution (parameterized by a multivariate normal distribution). By selecting the maximum value of the contact position marginal distribution across all actions (Eq. 1), a heat map representation can be generated to visualize the outcomes of the actions

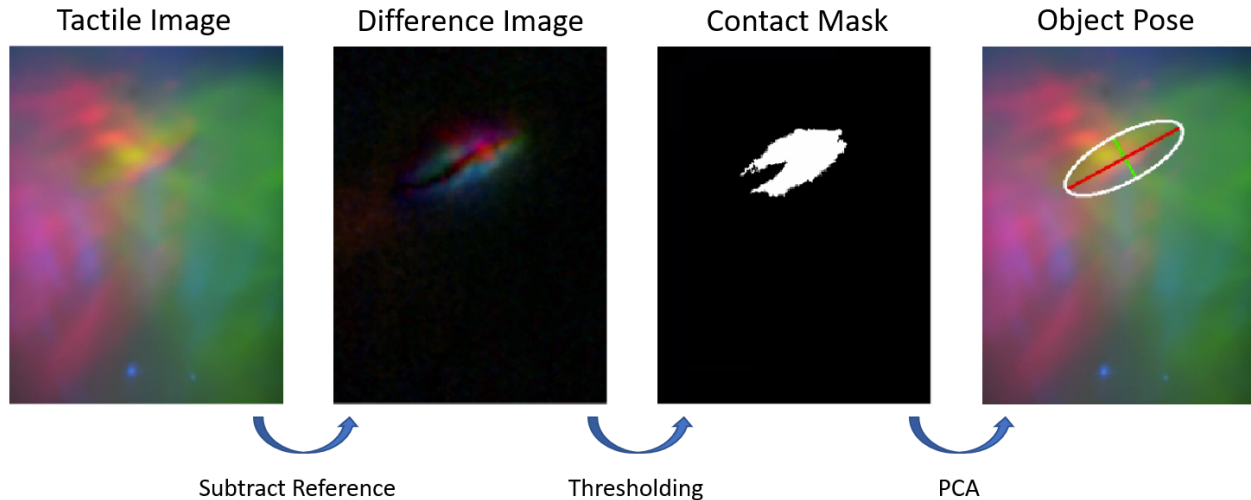


Fig. 3. The procedure to get the contact center position and orientation

tactile perception. The pipeline is shown in Fig. 3. Different from the method for cable manipulation [4], we adopted the color difference of the tactile image instead of the contact depth image for contact detection. The observation is that thresholding on the absolute value of the difference image is often as effective as thresholding on the contact depth for PCA (principal component analysis). Requiring no contact depth reconstruction also saves the calibration burden for vision-based tactile sensors [20], which is more complicated for the Digit sensor than the GelSight due to non-uniform illumination in different colors across sensors. Using the

color difference might introduce drawbacks for inconsistent contact detection at different locations on the tactile sensor, but we did not notice a significant impact in our experiments. The tactile perception produces four values (s, x, y, θ) . s is the mean value of the absolute image difference within the contact area between the tactile image and the no-contact reference image and serves as a contact strength indicator, i.e., how strong the current contact is. We observed the experimental statistics and chose to use a contact strength threshold of 1.8 to distinguish a stable grasp and an unstable one, which approximately corresponds to 0.4N normal force.

x, y and θ are the contact center and object angle calculated with PCA.

C. The Dynamics Model

An uncertainty-aware dynamics model [26] was employed to adapt to a low data regime and ensure a safe grasp outcome. We chose the MLP (multilayer perceptron) structure for its simple nature and universal fitting ability without the need of prior knowledge of the actual physics, making the method more generic. 16 MLPs were trained separately under negative log-likelihood loss with independent random initialization and shuffling. Then, their outputs were ensemble to obtain a single prediction and its uncertainty (covariance) to ensure only confident actions were executed. The contact location and orientation predictions also come with uncertainty, making it possible to plan and control with uncertainty for more accurate and generalizable results. The full loss (in $s, x, y,$ and θ) is incurred only when the next grasp is successful (stable). When the next grasp in the training dataset is a failed one, only the loss in s is backpropagated. This strategy helps to preserve the high uncertainty in the prediction of $x, y,$ and θ when the grasp fails and increases the system’s robustness. We set the multilayer perceptrons to have five layers with 32 hidden units for each and use ReLU activation functions. The dropout rate is set at 0.1. We also use 2 parameters to represent a single rotation to facilitate learning angle continuity [27]. The complete learned model of 16 MLPs has only 70K parameters. Compared with the complexity by Lambeta *et al.*, it is an order of magnitude less than 1.2M [25], yet achieves comparable control performance as will be shown in Sec.IV.

IV. EXPERIMENTS AND RESULTS

A. Hardware Setup and Data Collection

With 0.1 radians sampling step size, the original 8 DoFs joint angle action space is a vast space with around 2 billion discrete actions. However, action filtering retained only 424 actions to avoid the two fingertips being too far away to hold the stick in between. This indicates that action filtering through kinematics is a simple yet effective strategy to reduce non-feasible actions and make the data collection process more efficient. Although some plausible actions might be removed in this process, we will show in Sec. IV-E that the actions are still capable of regulating the position in the fingertips, using other possible nearby actions and the neural network’s interpolation capability. A wooden stick with a 2.5mm radius and 12.5cm length (shown as object 1 in Fig 4) was used to collect the training data. Its radius is much smaller than glass marbles [25]. In this case, because the fingertips need to stay within a closer range to maintain a stable grasp with the stick, more estimation and control accuracy are required for the manipulation. Only object 1 in Fig. 4 was used to collect the data that trained the dynamics model. The other 3 objects were used to test the model’s generalization ability (see Section IV-D). Initially, the configuration was similar to Fig. 1. The object was then

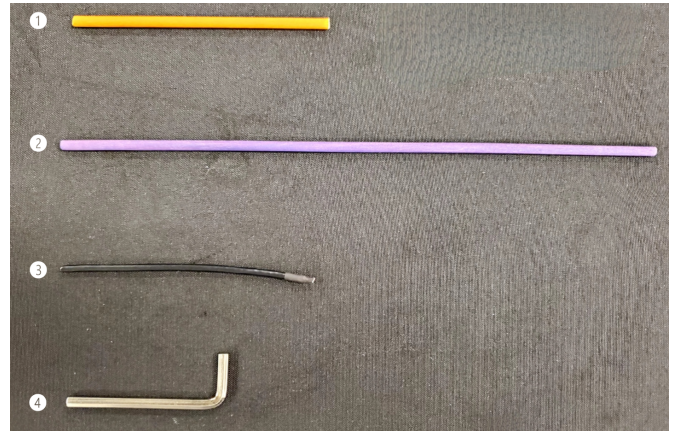


Fig. 4. Different objects used in the experiments. Only object 1 was used to collect the dataset for training the dynamics model. The other objects were used to test the generalization ability of the learned model

TABLE I
ABSOLUTE AND RELATIVE ERROR ON TEST DATA

	With Joint Torque	Without Joint Torque
Failure Strength	0.334(23.4%) \pm 0.017	0.364(25.4%) \pm 0.028
Success Strength	0.122(5.73%) \pm 0.006	0.130(6.10%) \pm 0.006
X Position (mm)	0.512(7.76%) \pm 0.0252	0.517(7.84%) \pm 0.0230
Y Position (mm)	0.570(8.39%) \pm 0.0474	0.629(9.26%) \pm 0.0409
Angle (Radian)	0.148(13.4%) \pm 0.008	0.147(13.3%) \pm 0.008

manipulated by randomly selecting actions from the 424 filtered choices to move the stick within the $305mm^2$ area on the fingertip, and the process continued indefinitely if the grasp of the object was maintained in the hand. When the grasp failed, the robotic hand was reset to a random state, and the object was manually put back into the hand. The tactile state (s, x, y, θ) and hand state (joint angle and torque) were recorded before and after the action. A total of 2250 state transitions were recorded during the data collection process, and the grasp failed in 361 of them after the sampled action was applied, resulting in a failure rate of 16%. This failure rate is significantly lower than that of random sampling in the original entire action space, as the robotic hand can quickly go into an unreasonable state. Because the actions are sampled uniformly, the resulting stick positions are also roughly evenly distributed among the fingertip. Compared with [25], which used 96000 state transitions to train, the data size in this work is much smaller. However, the approach we adopted collected the data and trained the model more wisely, thus making the method data efficient and scale better.

B. Training the Dynamics Model

The original 2250 sampled data was split into a training set of 1800 samples and a test set of 450 samples. Two different cases were compared when training the dynamics model of manipulating the object. One used the full hand state, including the joint angle feedback from the robotic

hand and the joint torque. The other was only using joint angle feedback as the hand state. This experiment is adopted to contrast the effectiveness of the hand torque proprioception in the presence of high-resolution tactile sensing. The evaluation of the neural network performance on test data is summarized in Table I. The results show that predicting the contact strength is more accessible when the next state is a successful grasp, as indicated by the lower prediction error. Although the contact strength prediction error for the failed cases is around 0.3, it approximately corresponds to only 0.06N in force. The successful grasp strength prediction is even more accurate. The position errors of both x and y axes are less than the millimeter level, which indicates that the neural network can predict the future object locations given the current object pose and the hand state. Interestingly, the performances with and without joint torque data do not show a significant distinction. A possible explanation is that tactile perception (the object’s pose and contact strength) already encodes the interaction information between the object and the robotic fingers, which is lower-dimensional and more informative than the hand torque data. Therefore, introducing the hand torque state to the input will provide little additional information to the system. Hence, the joint torque data will not be fed into the system in the remainder of this paper to reduce the neural network complexity.

C. Model Prediction Accuracy

The prediction accuracy of the learned dynamics model is also tested on real hardware. Specifically, we hope to test the model’s prediction ability on actions with different output confidence. To achieve this, the finger was initialized to grasp the trained object with a random pose. All the discrete actions from Section III-A were fed into the learned dynamics model to obtain the predicted contact strength, object pose, and corresponding confidence. To preserve only actions that would result in a stable grasp, all actions with contact strengths less than 1.8 (the threshold between a stable contact and an unstable one) with greater than 5% probability were filtered. Among the remaining actions, their resulting contact center location can be plotted as a probability heat map (as shown in Fig. 2) to visualize the positional distribution of the actions. The probability heat map was generated by taking the maximal probability among all the safe actions at each location in the tactile image. Mathematically, for a point $M(x, y)$ on the map M ,

$$M(x, y) = \max_{a \in \mathcal{A}} \Pr(x, y | a) \quad (1)$$

where \mathcal{A} is the set of the remaining actions. The maximum point in the heat map (brightest point in Fig. 2) corresponds to an action with the most confident resulting location. By selecting locations with different probability levels on the heat map and executing the actions behind these locations, we can test the dynamics model’s prediction behavior on actions with varying confidence levels. The values on the probability heat map are normalized to the range between 0 to 1, referred to as the confidence level in Table II for visualization and testing purposes. The hand chose the action

with the corresponding confidence level in each run, and the resulting contact location was observed. The experiment was repeated for 50 runs for each confidence level to get the average performance. It can be seen in Table II that the object did not fall out of the hand for all the cases. It is evident that filtering the actions by the contact strengths effectively produces a next state with safe and stable contact. The correlation between the magnitude of the errors and the actions’ confidence level is not very strong because although the confidence level relatively categorizes different actions for showing the statistics, each run has different absolute probabilities. Hence, the confidence level only serves to organize and present the experiment results. In practice, the absolute probability will be used to estimate the action errors.

TABLE II
ERROR STATISTICS ON REAL HARDWARE

Confidence Level	X Position Error (mm)	Y Position Error (mm)	Object Fall Frequency
1.0	0.302 ± 0.190	0.346 ± 0.234	0/50
0.9	0.507 ± 0.229	0.414 ± 0.370	0/50
0.8	0.439 ± 0.292	0.521 ± 0.366	0/50
0.7	0.307 ± 0.258	0.385 ± 0.283	0/50
0.6	0.575 ± 0.439	0.531 ± 0.375	0/50
0.5	0.595 ± 0.439	0.404 ± 0.322	0/50
0.4	0.712 ± 0.429	0.482 ± 0.317	0/50

For the prediction accuracy on the object’s angle, because only two fingers are employed during the manipulation, the predicted distribution of the change of the object orientation is unimodal and narrowly spread. Therefore, only the one action with the least standard deviation for output angle is executed during testing. The prediction error averaged over 50 runs is 0.078 ± 0.090 radians.

D. Generalization to Novel Objects

To test the dynamics model’s ability to generalize to other novel objects, in addition to the trained object (1# as shown in Fig. 4), three other objects were used to conduct the same experiment as in Section IV-C. Object 2# is the same as the trained object but 2 times longer. Object 3# is an electric wire piece representing a different material. Object 4# is a hex key of even more complicated inertial property. The results are summarized in Table III. In general, the performance is reduced with larger position errors and variance due to unseen objects. The results indicate that object 3# is the easiest to manipulate compared to the other two, as it falls out of the hand less frequently than the others. Regarding the objects’ physical properties, object 3# is most similar to the trained object in terms of radius, length, shape, etc. The longer length of object 2# makes it more difficult than object 3# as the inertia has changed. Object 4# is the most difficult to deal with, as can be anticipated, because it is heavier, thicker, and has an imbalanced weight. The positional errors and fall frequencies did not deteriorate much for all three objects when the confidence level was larger than 0.7. The reason might be that the most confident predictions are learned from the most basic physical properties of the interactions, and they tend to be more likely to generalize.

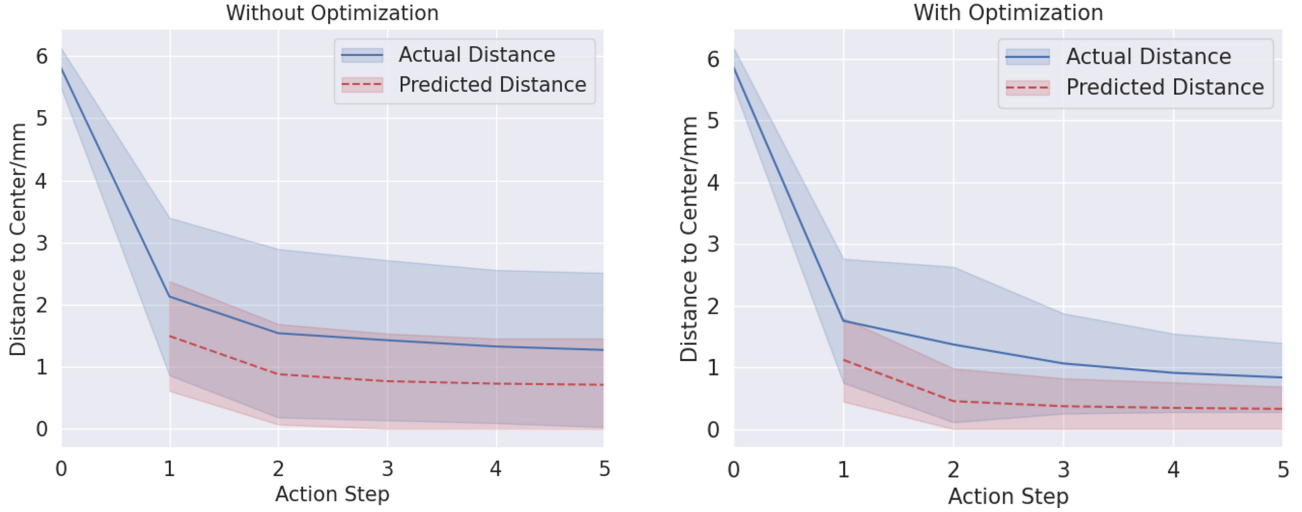


Fig. 5. Predicted and actual contact distances to the finger center by applying the best actions iteratively. On the right, a gradient-based optimization was added to improve the actions further

This dependency on fall frequency and prediction confidence level supports using the uncertainty-aware model. By predicting the model’s confidence using multiple MLPs instead of only one, we can overcome data depletion and overfit by choosing from confident actions with more stable behavior. The positional errors were kept within the millimeter level for most cases. This reveals that the dynamics model learned some generalizable features and was, to some extent, robust to the variation of the object property.

TABLE III
ERROR STATISTICS ON DIFFERENT OBJECTS

Confidence Level	X Position Error (mm)	Y Position Error (mm)	Object Fall Frequency
Object 2			
1.0	0.439 ± 0.093	0.302 ± 0.253	0/50
0.9	0.375 ± 0.463	1.062 ± 0.370	1/50
0.8	0.546 ± 0.361	0.517 ± 0.346	2/50
0.7	0.872 ± 0.590	0.453 ± 0.322	0/50
0.6	1.097 ± 0.609	0.653 ± 0.585	1/50
0.5	0.634 ± 0.443	0.487 ± 0.395	1/50
0.4	0.770 ± 0.531	0.336 ± 0.424	4/50
Object 3			
1.0	0.531 ± 0.336	0.439 ± 0.249	0/50
0.9	0.551 ± 0.565	0.453 ± 0.322	0/50
0.8	1.330 ± 0.590	0.551 ± 0.370	0/50
0.7	0.936 ± 0.565	0.595 ± 0.482	3/50
0.6	0.833 ± 0.356	0.453 ± 0.366	0/50
0.5	0.712 ± 0.429	0.434 ± 0.317	5/50
0.4	0.707 ± 0.419	0.653 ± 0.419	1/50
Object 4			
1.0	0.404 ± 0.322	0.707 ± 0.414	0/50
0.9	0.375 ± 0.249	0.750 ± 0.668	2/50
0.8	0.356 ± 0.297	0.838 ± 0.658	2/50
0.7	0.736 ± 0.468	0.575 ± 0.453	1/50
0.6	0.838 ± 0.507	0.614 ± 0.453	2/50
0.5	1.574 ± 0.863	0.936 ± 0.755	5/50
0.4	1.145 ± 0.877	1.330 ± 1.019	10/50

E. Application: Pose Regulation Experiment

Using the learned dynamics model, we can control the contact position of the object with the finger. For example,

the contact position can be adjusted to the exact center of the finger, given any initial configurations. In this experiment, we placed object 1# approximately 6 mm away from the center of the finger in a random position. All the valid actions from Section III-A were fed into the dynamic model to obtain the corresponding future predictions. Actions that resulted in unstable contact were first removed (same as in Section IV-C). Among the remaining ones, the action that gave a mean location prediction closest to the center of the finger was selected and executed. This process was then repeated for 5 steps to make the contact iteratively closer to the center. Another experiment examined that neural networks learned a smooth landscape in action space by using additional gradient-based optimization on the filtered discrete action choices. The optimization took the ℓ_2 norm of the distance from the finger center to the mean position prediction as the loss and backpropagated the gradient to the input action as the direction to improve the original action. Adam optimizer [28] was adopted, and the optimization was conducted for 100 iterations with a learning rate of 0.0005. Both experiments were conducted 30 times to obtain the expected performance over random initial positions. The predicted and actual distances from the contact center to the finger center in each step are shown in Fig. 5. The results demonstrate that both methods can regulate the contact to approach the finger center. It suggests that although keeping only the confident actions will reduce the number of reachable states in the state space, the model can still reach the desired final state, possibly at the expense of more motion steps but with uncertainty reduced and, therefore, safety improved. Moreover, the additional gradient-based optimization can quickly reduce the distance within the millimeter level. The improvement with the optimization also suggests that the learned dynamics model is capable of predicting the outcome of the original discrete actions and exhibits good performance on the continuous action space,

making it possible to generalize to unseen actions owing to the neural network's interpolation ability.

We compare this result with [25], where a marble is manipulated instead of a stick-like object. The marble has simpler dynamics because its radius is larger than the stick, and the mass is more concentrated; therefore, the marble has a larger tolerance on the control accuracy. Nonetheless, we achieved comparable accuracy in controlling the stick with much less data and model complexity than on the marble. Moreover, our learned model is uncertainty aware, making the actions less likely to drop the object. The stick was never dropped out of the hand in our experiment, which was not achieved in the prior work.

V. CONCLUSION

This work presented a data-efficient framework for learning to manipulate a small stick with high accuracy on real hardware. The approach involves reducing the feasible action space, leveraging model-based tactile signal extraction, and utilizing an uncertainty-aware contact dynamics model. The experimental results demonstrate that the framework achieves sub-millimeter precision. The learned model also has simpler complexity and requires less data than similar works. More importantly, by incorporating the uncertainty-aware model, we significantly reduced the probability of object fall, although the stick is even more challenging than the marble to manipulate due to additional dynamics complexity. Using the learned contact prediction model, our robotic hand can control the rolling object with precision and safety.

REFERENCES

- [1] N. C. Daffe, A. Rodriguez, R. Paolini, B. Tang, S. S. Srinivasa, M. Erdmann, M. T. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge, "Extrinsic dexterity: In-hand manipulation with external forces," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 1578–1585.
- [2] C. Wang, S. Wang, B. Romero, F. Veiga, and E. Adelson, "Swingbot: Learning physical features from in-hand tactile exploration for dynamic swing-up manipulation," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5633–5640.
- [3] T. Bi, C. Sferrazza, and R. D'Andrea, "Zero-shot sim-to-real transfer of tactile control policies for aggressive swing-up manipulation," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5761–5768, 2021.
- [4] 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/10.1177/027836492111027233>
- [5] M. Matak and T. Hermans, "Planning visual-tactile precision grasps via complementary use of vision and touch," *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 768–775, 2022.
- [6] A. Handa, A. Allshire, V. Makoviychuk, A. Petrenko, R. Singh, J. Liu, D. Makoviychuk, K. Van Wyk, A. Zhurkevich, B. Sundaralingam *et al.*, "Dextreme: Transfer of agile in-hand manipulation from simulation to reality," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5977–5984.
- [7] K. Shimonomura, "Tactile image sensors employing camera: A review," *Sensors*, vol. 19, no. 18, p. 3933, 2019.
- [8] T. Ishihara, A. Namiki, M. Ishikawa, and M. Shimojo, "Dynamic pen spinning using a high-speed multifingered hand with high-speed tactile sensor," in *2006 6th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2006, pp. 258–263.
- [9] K. Shaw, A. Agarwal, and D. Pathak, "LEAP Hand: Low-Cost, Efficient, and Anthropomorphic Hand for Robot Learning," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.
- [10] O. M. Andrychowicz, B. Baker, M. Chociej, R. Józefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020. [Online]. Available: <https://doi.org/10.1177/0278364919887447>
- [11] I. Popov, N. Heess, T. Lillicrap, R. Hafner, G. Barth-Maron, M. Vecerik, T. Lampe, Y. Tassa, T. Erez, and M. Riedmiller, "Data-efficient deep reinforcement learning for dexterous manipulation," *arXiv preprint arXiv:1704.03073*, 2017.
- [12] F. Chen Chen, S. Appendino, A. Battezzato, A. Favetto, M. Mousavi, and F. Pescarmona, "Constraint study for a hand exoskeleton: human hand kinematics and dynamics," *Journal of Robotics*, vol. 2013, 2013.
- [13] G. Khandate, C. Mehlman, X. Wei, and M. Ciocarlie, "Value guided exploration with sub-optimal controllers for learning dexterous manipulation," *arXiv preprint arXiv:2303.03533*, 2023.
- [14] S. P. Arunachalam, S. Silwal, B. Evans, and L. Pinto, "Dexterous imitation made easy: A learning-based framework for efficient dexterous manipulation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5954–5961.
- [15] K. Shaw, S. Bahl, and D. Pathak, "Videodex: Learning dexterity from internet videos," in *Conference on Robot Learning*. PMLR, 2023, pp. 654–665.
- [16] Q. Chen, K. Van Wyk, Y.-W. Chao, W. Yang, A. Mousavian, A. Gupta, and D. Fox, "Learning robust real-world dexterous grasping policies via implicit shape augmentation," in *Conference on Robot Learning*. PMLR, 2023, pp. 1222–1232.
- [17] T. Chen, M. Tippur, S. Wu, V. Kumar, E. Adelson, and P. Agrawal, "Visual dexterity: In-hand dexterous manipulation from depth," in *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*, 2023.
- [18] H. Qi, B. Yi, S. Suresh, M. Lambeta, Y. Ma, R. Calandra, and J. Malik, "General in-hand object rotation with vision and touch," in *Conference on Robot Learning*. PMLR, 2023, pp. 2549–2564.
- [19] S. Dong *et al.*, "High-resolution tactile sensing for reactive robotic manipulation," Ph.D. dissertation, Massachusetts Institute of Technology, 2021.
- [20] 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.
- [21] 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.
- [22] 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.
- [23] I. H. Taylor, S. Dong, and A. Rodriguez, "Gelslim 3.0: High-resolution measurement of shape, force and slip in a compact tactile-sensing finger," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 10 781–10 787.
- [24] Y. Du, G. Zhang, and M. Y. Wang, "3d contact point cloud reconstruction from vision-based tactile flow," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 12 177–12 184, 2022.
- [25] M. Lambeta, P.-W. Chou, S. Tian, B. Yang, B. Maloon, V. R. Most, D. Stroud, R. Santos, A. Byagowi, G. Kammerer *et al.*, "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, 2020.
- [26] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.
- [27] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5745–5753.
- [28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.