AnyRotate: Gravity-Invariant In-Hand Rotation with Sim-to-Real Touch

Max Yang, Chenghua Lu, Alex Church, Yijiong Lin, Chris Ford, Haoran Li, Efi Psomopoulou, David A.W. Barton*, Nathan F. Lepora*

***Abstract*— We present AnyRotate, a system for gravity-invariant multi-axis in-hand object rotation using dense featured touch. In our work, We leverage explicit contact features in simulation to provide rich tactile feedback for training a policy with Reinforcement Learning. We perform zero-shot policy transfer to reality by training an observation model to bridge the sim-to-real gap. In the real world, we successfully demonstrated that our dense touch policy can generalize to multiple hand directions for rotating various objects in arbitrary rotation axes. Interestingly, we found that despite not having explicit slip detection, rich multi-fingered tactile sensing can implicitly detect object movement within grasp and provide a reactive behavior that improves the robustness of the policy, highlighting the importance of information-rich tactile sensing for in-hand manipulation.**

I. Introduction

The dexterity of the human hand is fundamental to our daily routines. From writing with a pen to opening a jar, the ability to manipulate various objects with different shapes, sizes, and materials has likewise been a long-standing goal for robot manipulation [1]. However, these tasks can be hugely challenging for robot hands due to the high degree of actuation, fine motor control, and large environmental uncertainties. Whilst significant advances have been made in recent years, most prominently the work by OpenAI [2], [3], they primarily have been relying on vision-based systems which are not necessarily well suited to this task due to significant self-occlusion. To overcome this issue, it often results in complicated setups involving multiple cameras that are not representative of natural embodiment.

In this paper, we present AnyRotate: a robot system that can perform multi-axis in-hand object rotation that is invariant to gravity direction (hand orientation) using only proprioception and touch. We formulate the problem as continuous in-hand object rotation via stable precision grasping without any supporting surface [4]. Our RL formulation uses an auxiliary subgoal curriculum to train a unified policy that can achieve object rotation about any arbitrary target rotation axis relative to the hand. We leverage the highly parallelized IssacGym simulator [5] and privileged information to train a policy using a two-stage teacher-student training strategy. To bridge the sim-to-real gap for tactile observations, we collect contact data in the real world and train an observation model to extract explicit contact features used in the simulation. We show that our methodology allows us to train on simple fundamental shapes in simulation but generalize to differently sized-objects in the real world.

All authors are with the School of Engineering Mathematics and Technology and Bristol Robotics Laboratory, University of Bristol, Bristol BS8 1UB, U.K. (email: {max.yang, chenghua.lu, alex.church, yijiong.lin, chris.ford, haoran.li, efi.psomopoulou, david.barton, n.lepora}@bristol.ac.uk)

* These authors contributed equally.

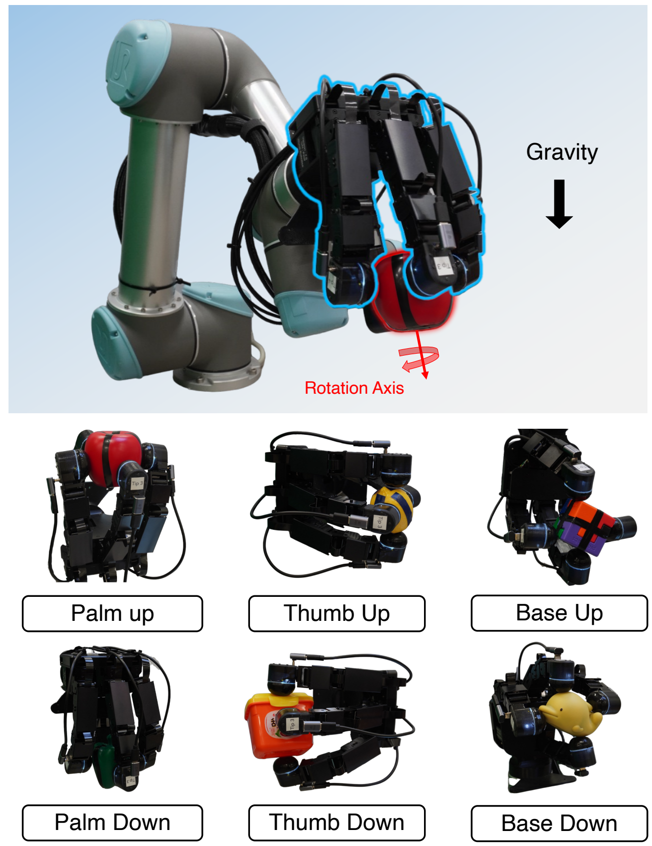

Fig. 1: Robot system setup: A UR5 with a 4-fingered 16-DoF tactile robot hand performing in-hand object rotation about any chosen axis and in different hand orientations with respect to gravity.

II. Method

A. Reinforcement Learning

We formulate a finite horizon goal-conditioned Markov Decision Process (MDP) defined by a continuous state $s \in \mathcal{S}$, a continuous action space $a \in \mathcal{A}$, a probabilistic state transition function $p(s_{t+1}|s_t, a_t)$, a reward function $r \in \mathcal{R} : \mathcal{S} \times \mathcal{A} \times \mathcal{G} \to \mathbb{R}$, and a goal space $\mathcal{G}$ with goal $g \in \mathcal{G}$.

*1) **Observation Space**:* The observation O_t contains the current and target joint position $q_t, \bar{q}_t \in \mathbb{R}^{16}$, previous action $a_{t-1} \in \mathbb{R}^{16}$, fingertip position $f_t^p \in \mathbb{R}^{12}$, fingertip orientation $f_t^r \in \mathbb{R}^{16}$, binary contact $c_t \in [0, 1]^4$, contact pose $P_t \in \mathbb{S}^4$, contact force magnitude $F_t \in \mathbb{R}^4$, and the desired rotation axis $\hat{k} \in \mathbb{S}^2$. The privileged information provided to the teacher policy includes object position, orientation, angular velocity, dimensions, gravity forces on the center of mass, and the current goal orientation.

*2) **Action Space**:* At each time step, the action output from the policy is the relative joint positions $a_t := \Delta\theta \in \mathbb{R}^{16}$. To encourage smooth finger motion, we apply an exponential moving average to compute the target joint positions. The target joint position is then defined as $\bar{q}_t = \bar{q}_{t-1} + \tilde{a}_t$, where $\tilde{a}_t = \eta a_t + (1-\eta)a_{t-1}$.

*3) **Auxiliary Goal**:* For training a unified policy for multi-axis rotations, we design a subgoal curriculum for more efficient training. We frame the object-rotation problem as object reorientation, where goals are placed about the rotation axis. The first target is generated by rotating the starting object orientation by a rotation increment about the rotation axis. Once the object reaches the target orientation, a new goal is generated by rotating the previous goal about the desired rotation axis by a fixed increment. The same operation is repeated until the episode ends. The rotation increment is defined as a regular interval.

*4) **Reward Design**:* We design a goal-based reward function to achieve continuous rotation while also avoiding inefficient hand motion.

$$r = r_{\text{rotation}} + r_{\text{contact}} + r_{\text{smooth}} + r_{\text{terminate}}, \qquad (1)$$

where:

$r_{\text{rotation}} = \lambda_{\text{kp}} r_{\text{kp}} + \lambda_{\Delta\text{rot}} r_{\Delta\text{rot}} + r_{\text{goal}}$,

$r_{\text{contact}} = \lambda_{\text{gc}} r_{\text{gc}} + \lambda_{\text{bc}} r_{\text{bc}}$,

$r_{\text{stable}} = \lambda_{\omega} r_{\omega} + \lambda_{\text{pose}} r_{\text{pose}} + \lambda_{\text{work}} r_{\text{work}} + \lambda_{\text{torque}} r_{\text{torque}}$,

$r_{\text{terminate}} = r_{\text{penalty}}$

The first part of the reward r_{rotation} is to maintain the continual object rotation objective. We use a keypoint formulation λ_{kp} to define the distance between the current and target pose [6]. We also add a sparse reward r_{goal} when a goal is reached and a delta rotation $r_{\Delta\text{rot}}$ to encourage continuous rotation about the target rotation axis. The morphology of the tactile sensor used here provides the most accurate tactile sensing information when the contact is normal to the fingertip. Therefore, to discourage the robot from leveraging other parts of the sensor, such as the tip edge or camera casing, we include a contact reward r_{contact} which rewards tip contacts r_{gc} and penalizes contacts with any other parts of the hand r_{bc}. In the third term of the reward function, we include a term to encourage stable rotations r_{stable} comprising: an object angular velocity penalty, r_{ω}; a hand-pose penalty on the distance between the joint position from a canonical pose, r_{pose}; a controller work-done penalty, r_{work}; and a torque penalty r_{torque}. Finally, we include an early termination penalty, $r_{\text{terminate}}$, which penalizes the agent ($\lambda_{\text{penalty}} = 50$) if the object falls out of the grasp or if the local rotation axis deviates too far from the global rotation axis.

B. Sim-to-Real Transfer

*1) **Teacher-Student Distillation**:* The policy trained in section II-A uses privileged information in simulation, such as object properties and auxiliary goal pose. To transfer this policy to reality, we use policy distillation to train a student policy that takes proprioception and tactile history as input to imitate the teacher policy. The student policy has the same actor-critic architecture as the teacher with the input $a_t = \pi_\theta(O_t, a_{t-1}, Z_t)$, where the latent vector $Z_t = \phi(O_t, O_{t-1}, ..., O_{t-n})$ is the predicted low dimensional encoding from a sequence of N proprioceptive and tactile observations. We use a temporal convolutional network (TCN) encoder for the latent vector function $\phi(x)$. We randomly initialize the student encoder network and initialize the policy network with the weights from the teacher policy. We train both the encoder and policy network via supervised learning with the mean squared error (MSE) of the latent vector and negative log-likelihood loss (NLL) of the action,

$$\mathcal{L} = \text{MSE}(z_t, \bar{z}_t) + \text{NLL}(a_t, \bar{a}_t) \qquad (2)$$

*2) **Tactile Feature Extraction**:* The tactile observations considered here are binary contact, contact pose, and contact force. We use a threshold on the SSIM between the current and the reference tactile image to calculate the binary contact signal. For bridging the sim-to-real gap of the remaining tactile observations, we adopt the sim-to-real framework from Ref. [7] where we use a trained observation model to perform zero-shot sim-to-real policy transfer. Given a real tactile image, we train a CNN model to extract the explicit features of contact force and contact pose. For contact pose, we use spherical coordinates to simplify contact definitions and use the contact pose variables polar angle R_x and azimuthal angle R_y relative to the fingertip origin. The predicted contact force variable is the total force magnitude of the contact. We also use the binary contact signal to mask the contact force and pose predictions.

III. Results and Conclusion

In the real world, we use the setup shown in 1. We compare three policies with different observation inputs, proprioception, binary touch (proprioception and binary contact), and dense touch (proprioception, binary contact, contact pose and contact force). We find that the policy trained with dense featured touch performed the best in the real world, demonstrating a successful sim-to-real transfer of our dense tactile representation. In noisy environments where the hand is in different orientations, a policy provided with tactile observations exhibited more stable rotating behavior and outperformed proprioception alone. In cases where the hand gets stuck at grasping the object firmly with not much rotation, an agent with tactile observation was able to increase finger movement to break contact.

While our investigation has focused on a specific case of in-hand manipulation using tactile sensing, we expect that the opportunities extend far beyond our demonstrations. The capability to manipulate objects effortlessly in free space with a sense of touch mirrors an intuitive skill of human dexterity. We hope that our research spurs continued efforts to reach a level of robot hand dexterity comparable to that of the human hand.

References

[1] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 1. IEEE, 2000, pp. 255–262.

[2] I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas *et al.*, "Solving rubik's cube with a robot hand," *arXiv preprint arXiv:1910.07113*, 2019.

[3] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.

[4] G. Khandate, S. Shang, E. T. Chang, T. L. Saidi, J. Adams, and M. Ciocarlie, "Sampling-based exploration for reinforcement learning of dexterous manipulation," *arXiv preprint arXiv:2303.03486*, 2023.

[5] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.

[6] A. Allshire, M. MittaI, V. Lodaya, V. Makoviychuk, D. Makoviichuk, F. Widmaier, M. Wüthrich, S. Bauer, A. Handa, and A. Garg, "Transferring dexterous manipulation from gpu simulation to a remote real-world trifinger," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 802–11 809.

[7] M. Yang, Y. Lin, A. Church, J. Lloyd, D. Zhang, D. A. Barton, and N. F. Lepora, "Sim-to-real model-based and model-free deep reinforcement learning for tactile pushing," *IEEE Robotics and Automation Letters*, 2023.