Active Perception for Tactile Sensing: A Task-Agnostic Attention-Based Approach

Tim Schneider[1], Cristiana de Farias[1], Roberto Calandra[2], Liming Chen[3], and Jan Peters[1,4]

***Abstract*— Humans make extensive use of haptic exploration to map and identify the properties of the objects that we touch. Also, in robotics, the use of active tactile perception has emerged as an important research domain that complements vision for tasks such as object classification, shape reconstruction, and manipulation. In this work, we introduce `TAP` (Task-agnostic Active Perception) – a novel framework that leverages reinforcement learning (RL) and transformer-based architectures to address the challenges posed by partially observable environments. `TAP` integrates Soft Actor-Critic (`SAC`) and `CrossQ` algorithms within a unified optimization objective, jointly training a perception module and decision-making policy. By design, `TAP` is task-agnostic and can, in principle, generalize to any active perception problem. We evaluate `TAP` across diverse tasks, including toy examples and a realistic application involving haptic exploration of 3D models of handwritten digits. Experiments demonstrate the efficacy of `TAP`, achieving a classification accuracy of 92% on Tactile MNIST. These findings underscore the potential of `TAP` as a versatile and generalizable framework for advancing active tactile perception in robotics.**

I. Introduction

Tactile perception enhances robotic manipulation tasks by complementing other sensing modalities to achieve classification, dexterous in-hand motions, and shape recognition. Unlike vision, which provides global scene understanding, tactile sensing is inherently local and requires active exploration to gather meaningful information [1]. This motivates the need for methods that can autonomously decide where and how to touch an object to maximize information gain.

Recent works have explored active tactile perception using heuristic-driven and supervised learning methods [2], [3]. However, many of these approaches are task-specific and do not generalize to new scenarios. In contrast, humans exhibit flexible exploration strategies, adapting their tactile interactions based on the context [4].

To bridge this gap, we introduce `TAP`, a framework that integrates reinforcement learning with a transformer-based architecture to enable adaptive tactile exploration. `TAP` jointly optimizes a policy for exploration and a perception model, allowing for efficient information acquisition. Crucially, `TAP` is fully task agnostic and could, in principle, be applied to any task with a differentiable loss function. We evaluate `TAP` on the Tactile MNIST benchmark, demonstrating its ability to learn structured exploration policies and achieve high classification accuracy of 92% on Tactile MNIST.

[1]Tim Schneider, Cristiana de Farias, and Jan Peters are with the Department of Computer Science, TU Darmstadt, Germany. {tim.schneider1, cristiana.farias, jan.peters}@tu-darmstadt.de

[2]Roberto Calandra is with LASR Lab & CeTI, TU Dresden, Germany. rcalandra@lasr.org

[3]Liming Chen is with LIRIS, CNRS UMR5205, Ecole Centrale de Lyon, France. liming.chen@ec-lyon.fr

[4]Jan Peters is also with the German Research Center for AI (DFKI), Hessian.AI, and the Centre for Cognitive Science, TU Darmstadt, Germany.

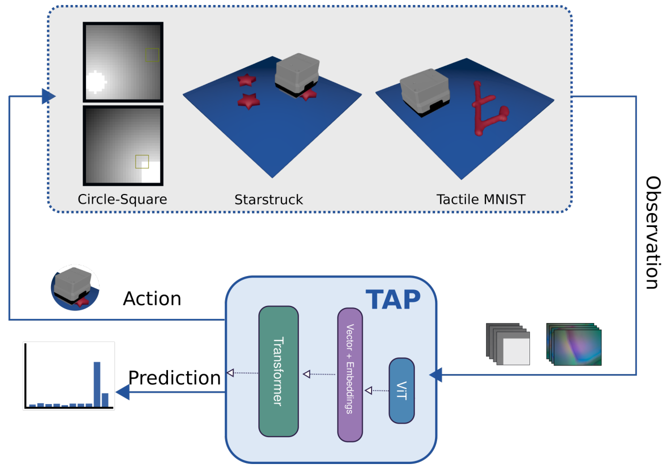

Fig. 1: Our method ***T**ask-agnostic **A**ctive **P**erception* (`TAP`). `TAP`'s objective is to infer properties, such as object classes, of its environment based on limited per-step information. To do so, it jointly optimizes an action policy to gather information and a prediction model for inference. Both the action policy and the prediction model use a shared transformer-based backbone to process sequences of inputs. Illustrated on the top are three benchmark tasks we use to evaluate `TAP`, though *Circle-Square* and *Starstruck* are omitted in the remainder of this paper for brevity.

II. Active Perception

Formally, we define the problem of active perception as a special case of a Partially Observable Markov Decision Process (POMDP) with hidden state h_t, action a_t, and observation o_t. In the active perception scenario, the agent's objective is to learn a particular property of the environment, e.g., the class or pose of an object. We assume that the ground truth value $\overset{*}{y}_t$ of this property at time t is part of the hidden state h_t and thus not directly accessible to the agent. To allow the agent to make predictions, its action space contains not only control actions a_t but also a current estimate y_t of the desired environment property. The resulting reward function now consists of two parts: a differentiable prediction loss ℓ and a regular RL reward r. The prediction loss could, for example, be a cross-entropy loss in the case of a classification task or the Euclidean distance in the case of a pose estimation task. The RL reward could be any function; in this work, we only use it to regularize the agent's actions a_t.

The objective is now to find a policy $\pi(a_t \,|\, o_{0:t})$ for which

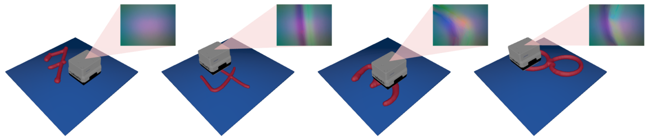

Fig. 2: The simulated *Tactile MNIST* classification benchmark [5]. The objective of the Tactile MNIST task is to identify the numeric value of the presented digit by touch only. The agent decides how to move the finger and predicts the class label of the presented MNIST digit. The haptic glance is computed via the Taxim [6] tactile simulator.

the expected discounted return is maximized:

$$\max_{\pi} J(\pi) \coloneqq \mathop{\mathbb{E}}_{p\left(\mathbf{h}, \overset{*}{\mathbf{y}}, \mathbf{o}, \mathbf{a}, \mathbf{y}\right)} \left[\sum_{t=0}^{\infty} \gamma^t \left(r(h_t, a_t) - \ell(\overset{*}{y}_t, y_t) \right) \right] \quad (1)$$

where $\gamma \in [0, 1)$ is the discount factor.

In this work, we assume that the policy π is parameterized by parameters $\theta \in \mathbb{R}^M$, which allows us to compute a gradient of (1) and optimize the problem with gradient descent algorithms.

Defining $\ell_\pi(\overset{*}{y}_t, o_{0:t}) \coloneqq \mathbb{E}_{\pi(y_t \mid o_{0:t})}\left[\ell(\overset{*}{y}_t, y_t)\right]$ and computing the gradient of $J(\pi_\theta)$ now yields

$$\frac{\partial}{\partial \theta} J(\pi_\theta) = \underbrace{\mathop{\mathbb{E}}_{p_\theta\left(\mathbf{h}, \overset{*}{\mathbf{y}}, \mathbf{o}, \mathbf{a}\right)} \left[\frac{\partial}{\partial \theta} \ln \pi_\theta(\mathbf{a} \mid \mathbf{o}) \sum_{t=0}^{\infty} \gamma^t \hat{r}(h_t, \overset{*}{y}_t, a_t, y_t) \right]}_{\text{policy gradient}}$$
$$- \underbrace{\mathop{\mathbb{E}}_{p_\theta\left(\overset{*}{\mathbf{y}}, \mathbf{o}\right)} \left[\sum_{t=0}^{\infty} \gamma^t \frac{\partial}{\partial \theta} \ell_{\pi_\theta}(\overset{*}{y}_t, o_{0:t}) \right]}_{\text{prediction loss gradient}}. \quad (2)$$

As shown in (2), the gradient of the objective function $J(\pi_\theta)$ decomposes into a policy gradient and a negative supervised prediction loss gradient.

We use RL-based techniques to estimate the policy gradient in Eq. (2), focusing on two actor-critic methods: `SAC` [7] and `CrossQ` [8]. To use `SAC` and `CrossQ` in the active perception setting, we adjust three components:

1) The active perception setting is partially observed. Hence, instead of a state s_t, the policy and the Q-networks receive a trajectory of past observations $o_{0:t}$.

2) The presence of the prediction loss ℓ_{π_θ} must be considered during the training of the Q-networks, yielding

$$\mathcal{L}_{\text{critic}} = \mathop{\mathbb{E}}_{\mathcal{D}} \left[\frac{1}{2} \left(Q_\theta(o_{0:t}, a_t) - \left(r_t - \ell_{\pi_\theta}(\overset{*}{y}_t, o_{0:t}) \right.\right.\right. \swarrow$$
$$\left.\left.\left. + \gamma \mathop{\mathbb{E}}_{\pi_\theta}[Q_{\bar{\theta}}(o_{0:t+1}, a_{t+1})] \right) \right)^2 \right].$$

3) During the policy update, the policy gradient is augmented by the prediction loss gradient.

For the remainder of this paper, we will refer to the `SAC` variant as `TAP-SAC` and to the `CrossQ` variant as `TAP-CrossQ`.

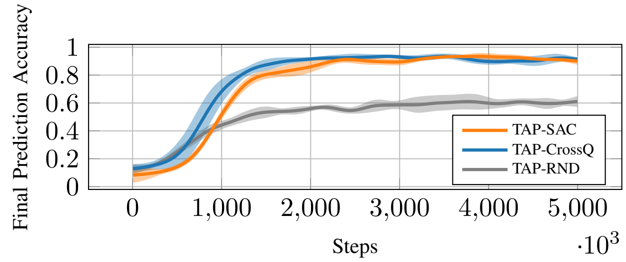

Fig. 3: Final prediction accuracies over five runs for our methods and the random baseline `TAP-RND` over the course of the training on the Tactile MNIST benchmark. The metrics displayed here are computed on the evaluation task, which contains objects unseen during the training.

We assume that the sequence of past observations $o_{0:t}$ consists of images and scalar data. To efficiently process this data in the policy and Q-networks, we use a Transformer on top of a Vision Transformer to compute an embedding of the observation sequence similar to [9]. We empirically found that sharing these embeddings across Q-networks, action policy, and prediction policy yields better results than training individual representations for each of these components. An overview of `TAP` is shown in Fig. 1.

III. Experiments and Results

We evaluate `TAP` on the Tactile MNIST benchmark [5], where an agent classifies 3D-printed digits by touch alone (see Fig. 2). The agent explores a 12x12 cm plate using a simulated GelSight sensor, taking up to 16 steps per episode. Each step provides a new tactile observation, and the agent predicts the digit's label throughout the episode.

We compare `TAP` against a random exploration baseline (`TAP-RND`), which moves the sensor randomly while training the perception module. As a metric, we measure the final prediction accuracy, which is the classification accuracy at the final exploration step of each episode. As shown in Fig. 3, both `TAP` methods achieve a final classification accuracy of 92%, significantly outperforming `TAP-RND` (60%), highlighting the importance of active exploration. Our RL-based approach enables structured exploration, reducing the number of steps required to make confident predictions.

IV. Conclusion and Future Work

We presented `TAP`, a reinforcement learning-based approach to active tactile perception, leveraging transformer-based architectures for joint policy and perception optimization. Our experiments on the Tactile MNIST benchmark demonstrate `TAP`'s ability to learn efficient exploration strategies, achieving high classification accuracy.

Future work includes extending `TAP` to real-world applications, incorporating multi-modal sensor fusion (e.g., vision and touch), and improving sample efficiency through pretraining techniques. Our results suggest that reinforcement learning and transformer models hold promise for advancing general-purpose active tactile perception in robotics.

REFERENCES

[1] Q. Li, O. Kroemer, Z. Su, F. F. Veiga, M. Kaboli, and H. J. Ritter, "A review of tactile information: Perception and action through touch," *IEEE Transactions on Robotics*, vol. 36, no. 6, pp. 1619–1634, 2020.

[2] A. Boehm, T. Schneider, B. Belousov, A. Kshirsagar, L. Lin, K. Doerschner, K. Drewing, C. Rothkopf, and J. Peters, "What Matters for Active Texture Recognition With Vision-Based Tactile Sensors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[3] J. Xu, S. Song, and M. Ciocarlie, "Tandem: Learning joint exploration and decision making with tactile sensors," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 391–10 398, 2022.

[4] R. S. Johansson, "Sensory input and control of grip," in *Novartis Foundation Symposium*. Wiley Online Library, 1998, pp. 45–58.

[5] T. Schneider, C. de Farias, R. Calandra, L. Chen, and J. Peters, "The Tactile MNIST Benchmark," https://sites.google.com/robot-learning.de/tactile-mnist, 2025.

[6] Z. Si and W. Yuan, "Taxim: An example-based simulation model for gelsight tactile sensors," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2361–2368, 2022.

[7] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[8] A. Bhatt, D. Palenicek, B. Belousov, M. Argus, A. Amiranashvili, T. Brox, and J. Peters, "CrossQ: Batch Normalization in Deep Reinforcement Learning for Greater Sample Efficiency and Simplicity," *arXiv preprint arXiv:1902.05605*, 2019.

[9] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6836–6846.