Supplementary for Learning Active Force-torque based Policy for Sub-mm Localization of Unseen Holes

Liang Xie[✉], **Haodong Zhang, Yinghao Zhao, Yue Wang**[✉], **Rong Xiong**[✉]

State Key Laboratory of Industrial Control Technology and Institute of Cyber-Systems and Control, Zhejiang University, Zhejiang, China

1. Introduction

In this document, we provide the details mentioned in the main paper, including the description of the transformer, the control mode, and the performance on different peg and hole materials.

A. Transformer

Consider two grid maps, the dense one X_d and the sparse one X_s, containing $N \times N$ grids $\mathbf{d}$. Each grid of the maps simultaneously flows through the Transformer. As the Transformer itself does not have any sense of positional order, we define positional encoding $\mathbf{p}$ to retain the spatial structure of the maps:

$$\mathbf{p}(k,2i) = sin\frac{k}{10000^{2i/d}} \tag{1}$$

$$\mathbf{p}(k,2i+1) = cos\frac{k}{10000^{2i/d}} \tag{2}$$

where k indicates the position of the grid, i is the dimension, and d is the input dimension of the Transformer. The position encoding $\mathbf{p}$ then adds to the embedded grid $\mathbf{d}$, which is transformed into a high-dimensional vector by a MLP:

$$\mathbf{x} = \mathbf{p} + \mathrm{MLP}(\mathbf{d}) \tag{3}$$

We adopt the attention mechanism to streamline the encoded map $\mathbf{x}$. The input vectors for an attention layer are named query, key, and value. The attention layer maps a query and a set of key-value pairs to an output. Concretely, the output is computed as a weighted sum of the values V. We obtain the attention weight from the dot product of the query Q and the key vector K corresponding to each value V, then apply a softmax function on the weights. Formally, the attention layer is denoted as:

$$\mathrm{Attention}(Q,K,V) = softmax(QK^T)V \tag{4}$$

For the self-attention layer, we perform intra-map attention on the dense and sparse maps. The input vectors are obtained by linear transformation:

$$Q = \mathbf{W}_{\mathrm{q}}\mathbf{x},\ K = \mathbf{W}_{\mathrm{k}}\mathbf{x},\ V = \mathbf{W}_{\mathrm{v}}\mathbf{x} \tag{5}$$

For the cross-attention layer, we perform the inter-map attention across the streamlined dense and sparse maps. We take each grid in the dense map as a query to retrieve attention (similarity) from the current grid in the sparse map. By conducting the self- and cross-attention in order, the locator learns to match from the sparse map to the dense map for contact point localization. The intuition behind this is that self-attention helps each map grid to be aware of the surrounding structure in their map for map matching. Then the cross-attention helps the current contact grid to be aware of the surrounding structure in the dense map for localization.

B. Control Mode

We choose the discrete position control instead of the continuous speed control because the measured forces and torques include the friction forces and torques, which are produced in the sliding between the peg-hole surfaces. In continuous control, the peg could slide in different directions and cause different friction forces and torques. Hence, the measured forces and torques will differ from the static ones in the pre-collected dense map, leading to failure in the following localization and matching. To solve the problem, we control the peg above the hole surface without contact at each time step of the position control. The robot starts by receiving the actions produced by the policy and outputting the torque commands to move the peg to the desired position. Then the contact forces and torques are measured and recorded as the observation of the next time-step. Finally, a small upwards

Table 1. *Policy Success Rate on Different Peg-hole Materials*

Success Rate	**Peg-hole Material**			
	metal peg +metal hole	plastic(pc) peg +metal hole	plastic(abs) peg +plastic(abs) hole	rubber peg +metal hole
10-step	21/25	23/25	24/25	21/25
20-step	24/25	25/25	25/25	23/25

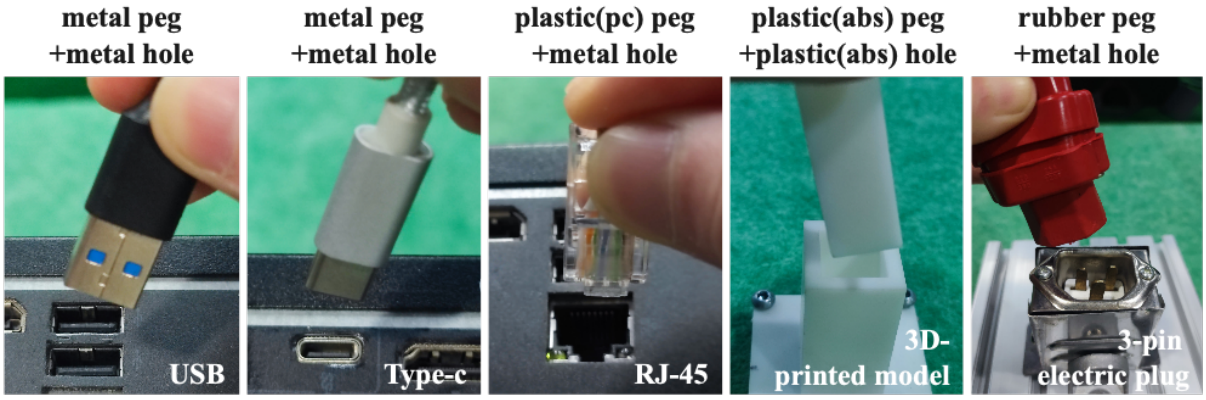

Fig. 1. The evaluated peg-hole pairs with different materials and elasticity.

move is initiated to keep the peg apart from the hole surface and prepare for the next control cycle. The videos of the experiments are available at https://github.com/xieliang555/Tracer for a more intuitive understanding. We use this control mode to minimize the uncertainty introduced by the friction forces and torques. Although the insertion efficiency and fluency decline to a certain extent by adopting this control mode, we find promising results in the generalization potential to unseen peg-hole pairs in the real world.

C. Performance on Different Materials

In the main manuscript, we have conducted experiments with the 3D-printed peg-hole models and various tight connector-socket pairs (e.g., USB, Type-C, C13, RJ-45). The materials of those pairs are very different. For example, the USB and Type-C mating parts are made of rigid metal. The RJ-45 (PC plastic) and the 3D-printed models (ABS plastic) are made of plastic materials with higher elasticity. Based on preliminary experiments, we further conduct experiments with a rubber connector made up of slightly deformable materials. We combine the peg-hole pairs with different materials, such as metal peg+metal hole (USB, Type-C), PC plastic peg+metal hole (RJ-45), ABS plastic peg+ABS plastic hole (3D-printed models), and rubber peg+metal hole (3-pin electric plug) as shown in Fig.1. The experiment results are summarized in Tab.1. We can find that our method can generalize to peg-hole pairs with different materials and elasticity. The experiment videos are available at https://github.com/xieliang555/Tracer