Learning Multiple Peg-in-hole Assembly Skills Using Multimodal Representations and Impedance Control

Wenkai Chen[1*], Chao Zeng[1], Fuchun Sun[2] and Jianwei Zhang[1], Presented by Wenkai Chen
[1] *Dept. of Informatics, University of Hamburg, Germany*
[2] *Dept. of Computer Science and Technology, Tsinghua University, China*

Abstract

Previously limited solutions for multiple peg-in-hole assembly are hard to transfer into real industrial scenarios flexibly. To effectively address these issues, this work designs a novel and more challenging multiple peg-in-hole assembly setup. Based on it, multiple modalities including vision, proprioception, and force/torque are learned as compact representations to account for the complexity and uncertainties of the environment. Furthermore, RL is used in the simulation to train the policy and the learned policy is transferred to the real world without extra exploration. Domain randomization and impedance control are embedded into the policy to narrow the gap between simulation and reality. Finally, the evaluation results demonstrate that the proposed solution can achieve successful multiple peg-in-hole assembly with the ability to generalize over different object shapes in the real world. Experimental video can be seen: https://github.com/turbohiro/Assembly_MPA

Methodology

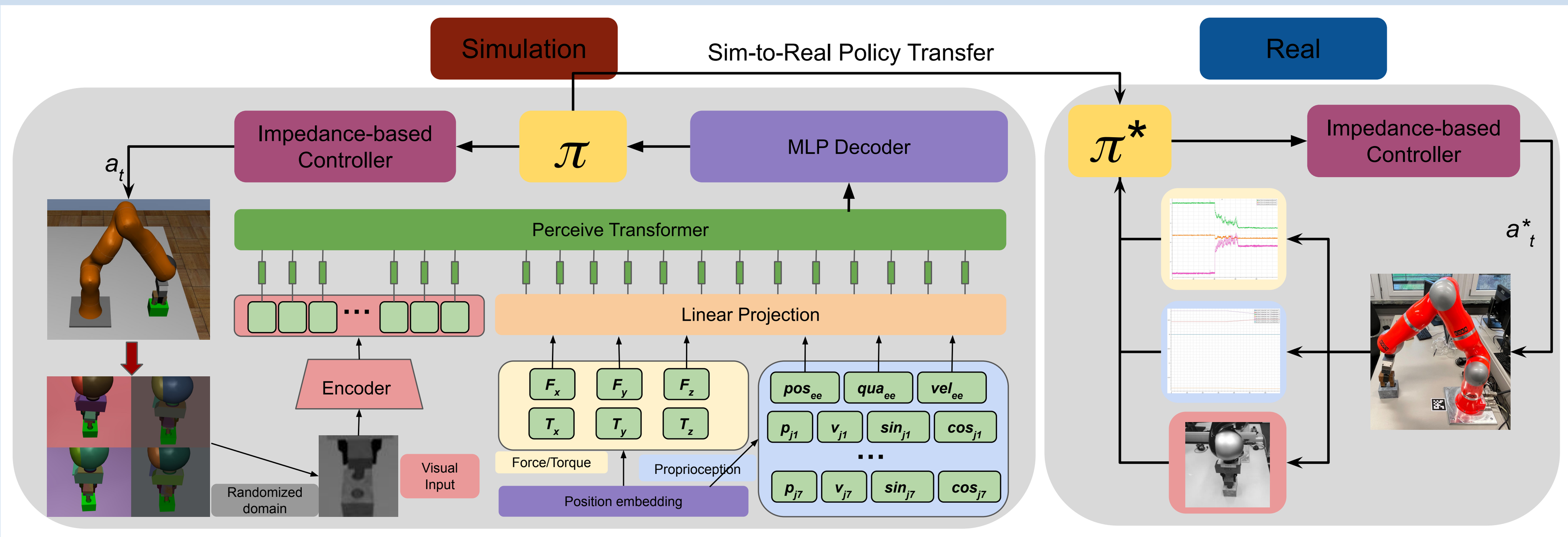

Figure 1: Overview of the proposed architecture. The left part represents policy training in the simulation environment, where domain randomization is used to sample the interaction of the assembly task like color, lighting, camera, and robotic dynamics (left). During the training process, multiple modalities including visual image, proprioception, and force/torque signals are all tokenized and fused into a perceived transformer module. Each predicted policy π is embedded into an impedance controller to execute torque commands to control the manipulator. Finally, the trained policy π^* is transferred to the real world without additional exploration (right).

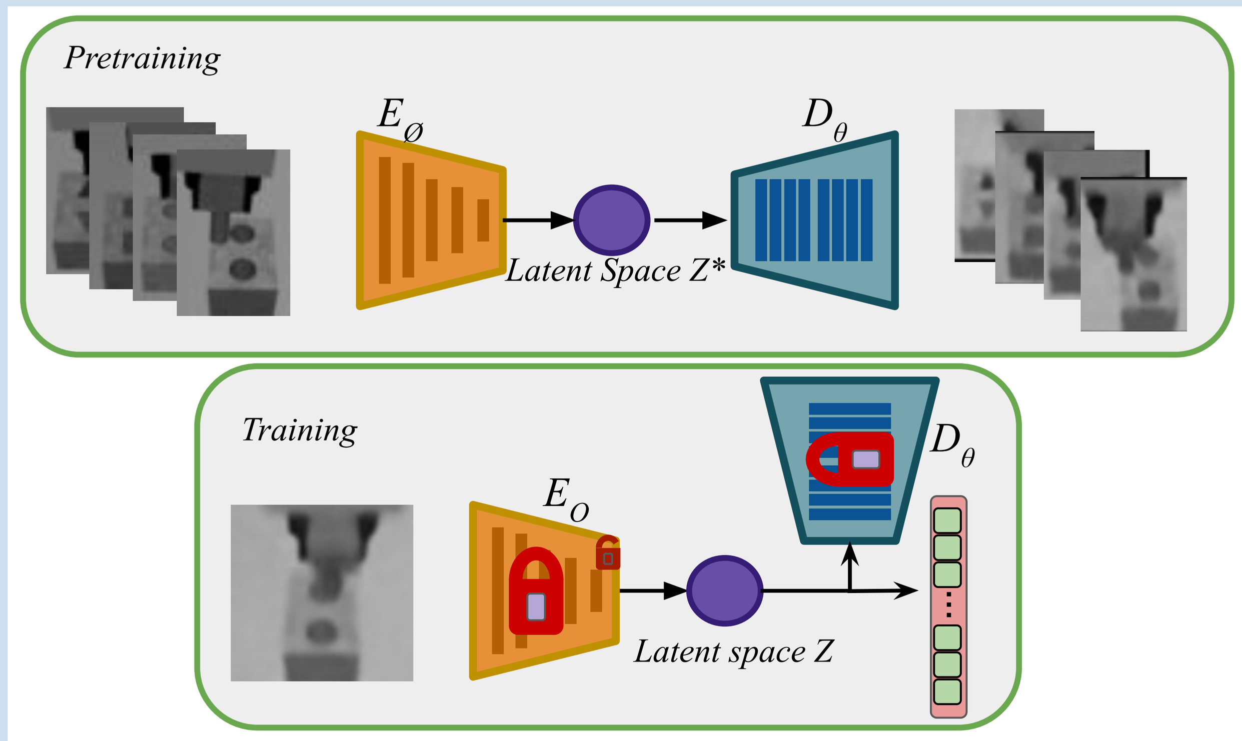

Figure 2: Overview of the visual representation module. 1) pretraining: We first pre-train the feature encoder and decoder. After that, the latent space Z^* to recognize different object shapes is obtained. 2) training: the decoder and most encoder layers are frozen, and we train the encoder during the policy learning process.

(1) The predicted action a_t is defined as:

$$a_t^{pos} = s_{t+1}^{pos} - s_t^{pos}, a^{pos} \in [\Delta x, \Delta y, \Delta z] \tag{1}$$

$$a_t^{ori} = s_{t+1}^{ori} * inv(s_t^{ori}), a^{ori} \in [\Delta\alpha, \Delta\beta, \Delta\gamma] \tag{2}$$

$$a_t^{gri} = s_{t+1}^{gri} - s_t^{gri}, a^{ori} \in [0, 1] \tag{3}$$

where t is the timestamp of the current action, and s^{pos}, s^{ori}, and s^{grpi} represent the scalar value for position, orientation, and gripper.

(2) After obtaining action a_t during each simulation step, the impedance controller framework computes the necessary joint torques to minimize the error between the desired and the current pose according to specified impedance parameters and torque limitations. Furthermore, SAC is used to train the multi-modality impedance-based assembly policy.

Results

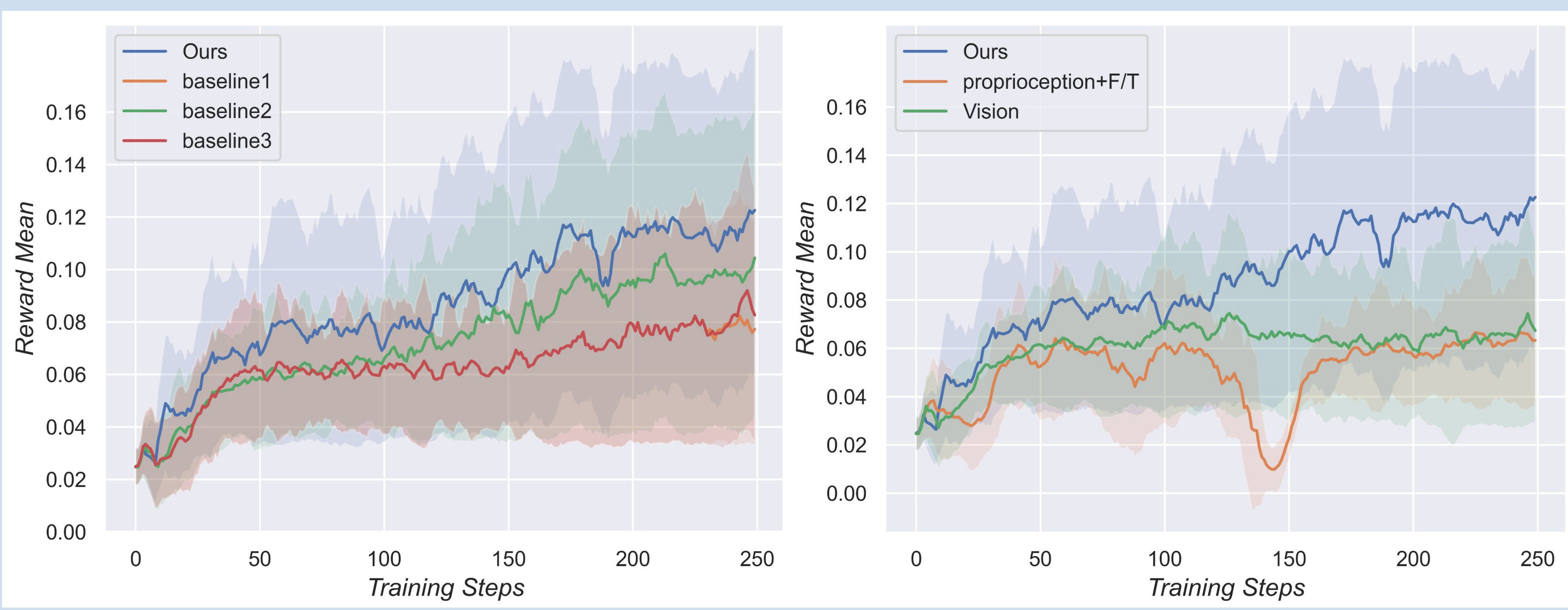

Figure 3: Ablation study for the proposed architecture .

Table 1: The comparison of final success rate and assembly time.

Object shape	Circle	Ellipse	Square	Triangle*	Triangle
Time (s)	15.0	18.0	18.0	24.2	17.6
Success rate % avg (std)	90.9 (9.09)	77.3 (13.6)	81.8 (9.09)	54.5 (9.09)	81.8 (9.09)

Note: the untrained object shape is represented with *.

Table 2: The comparison of final success rate considering the effect of domain randomization and impedance control.

Object shape	Circle	Ellipse	Square	Triangle*	Triangle
w/o domain randomization % avg (std)	13.6 (9.09)	9.09 (9.09)	0.0 (0.0)	0.0 (0.0)	0.0 (0.0)
w/o impedance controller % avg (std)	63.6 (13.6)	54.5 (9.09)	50.0 (9.09)	22.7 (9.09)	45.5 (13.6)

Note: the untrained object shape is represented with *.

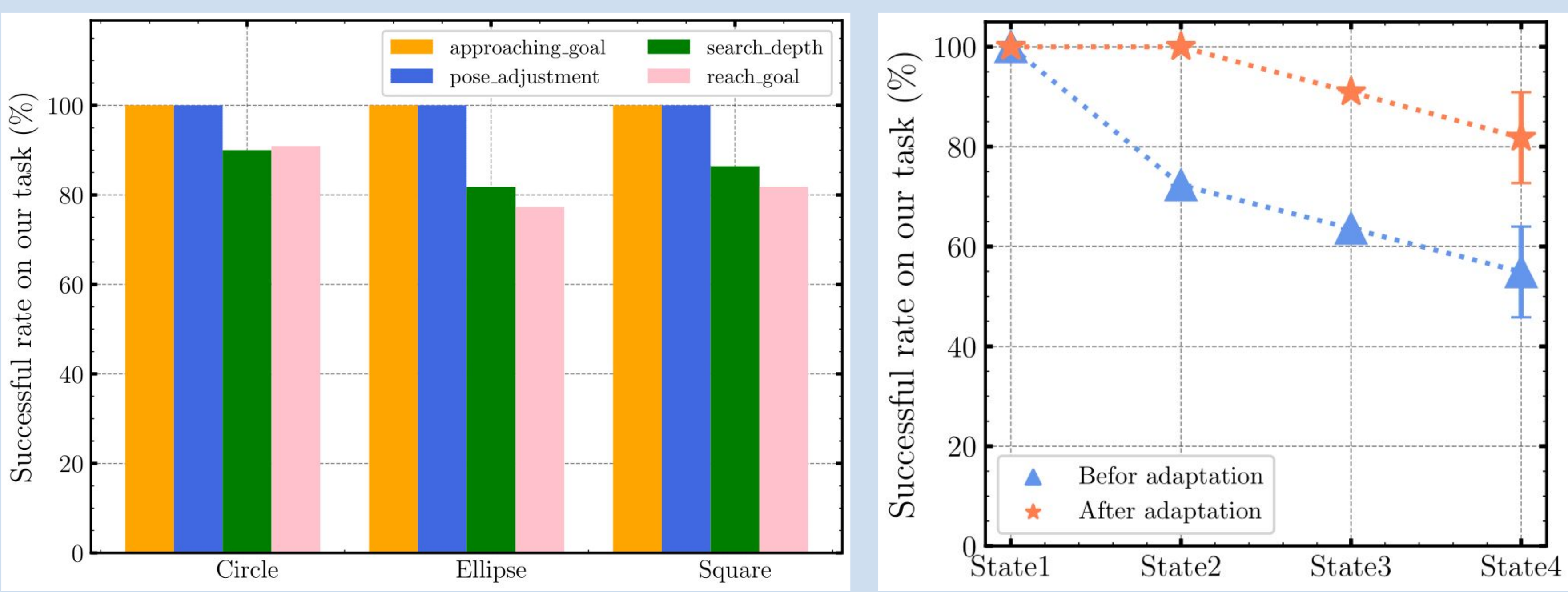

Figure 4: Comparison of the success rate of our task for different transition states from Circle, Ellipse, and Square objects (left) Triangle object (right).

Figure 5: Example of the proposed solution on multiple peg-in-hole assembly for different trained object shapes (a) circle (b) ellipse (c) square.

References

[1] Michelle A Lee, Yuke Zhu, Peter Zachares, Matthew Tan, Krishnan Srinivasan, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Learning multimodal representations for contact-rich tasks. *IEEE Transactions on Robotics*, 36(3):582–596, 2020.

[2] Jing Xu, Zhimin Hou, Wei Wang, Bohao Xu, Kuangen Zhang, and Ken Chen. Feedback deep deterministic policy gradient with fuzzy reward for robotic multiple peg-in-hole assembly tasks. *IEEE Transactions on Industrial Informatics*, 15(3):1658–1667, 2018.