Making Sense of Vision and Touch

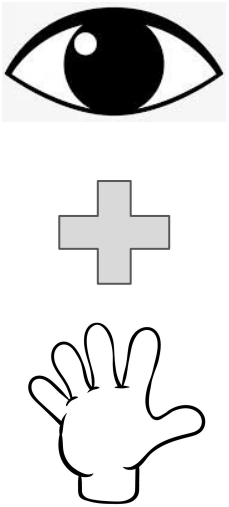
Self-Supervised Learning of Multimodal Representations for Contact-Rich Tasks

Presenter: Matthew Kozlowski

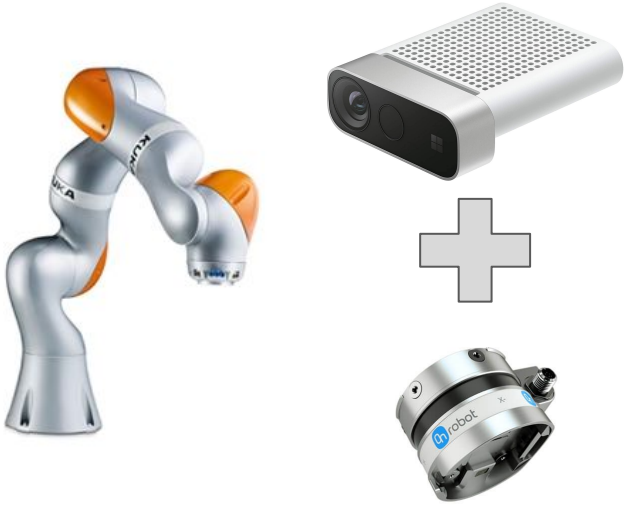
9/12/2023

What is Multimodal Manipulation?

Humans use sight and touch seamlessly in tasks



Goal: Enable robots to do the same



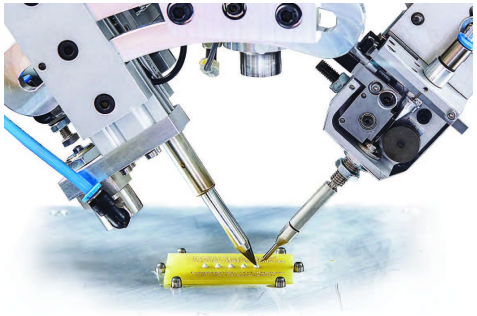
Why is Multimodal Manipulation Important?

It makes robots more human-like

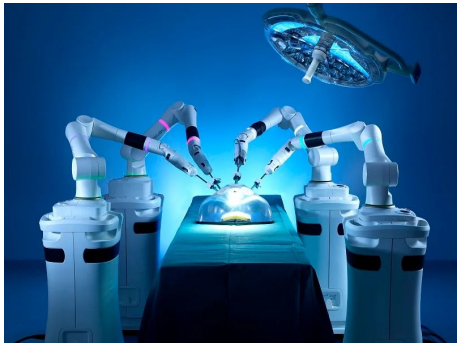
Improves their ability to reason about the world

Easier to achieve human-level performance on some tasks

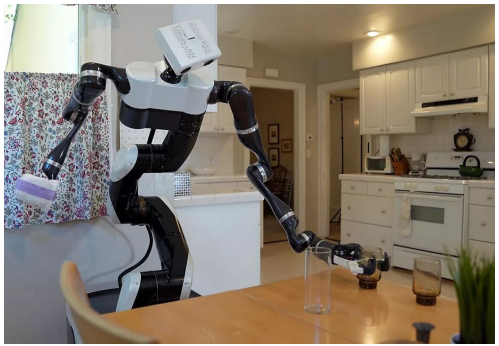
Potential Tasks and Applications



Manufacturing



Surgery



Household Tasks

Challenges with Multimodal Manipulation

It can be difficult to apply standard machine learning techniques, which normally expect a single type of data, to data of varying modalities

Natural challenges when working with real-time data:
noisy data, real-time processing speed requirements, adapting to changing environment

Problem Setting

- Objectives:
  - Learn neural-net based feature representation of sensory data
  - Learn a policy to perform a manipulation task through reinforcement learning

Maximize the reward:

$$J(\pi) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{T-1} \gamma r(\mathbf{s}_t, \mathbf{a}_t)\right]$$

Policy - neural net

States - learned data representation

Actions - 3D displacements in space

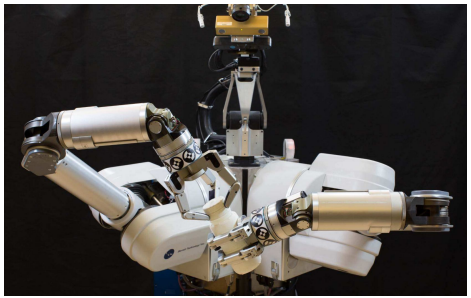
Related Work

Data-Driven Online Decision Making for Autonomous Manipulation

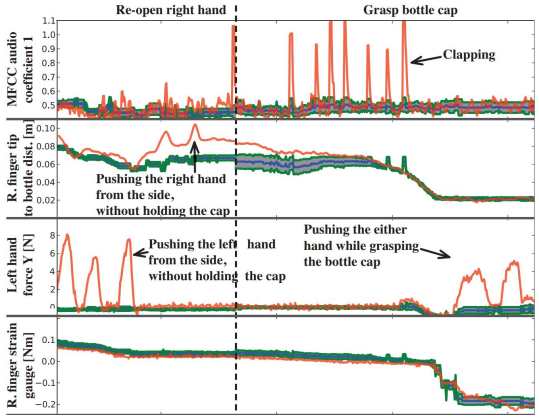
(Kappler et al., 2015)

Similarities

Manipulation Task



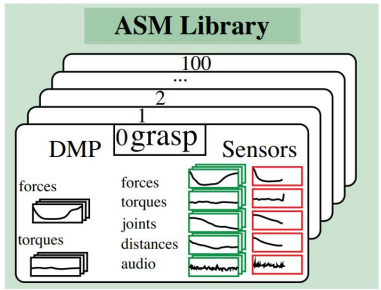
Multimodal Inputs

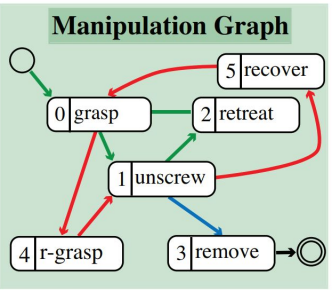


Limitations

Single task

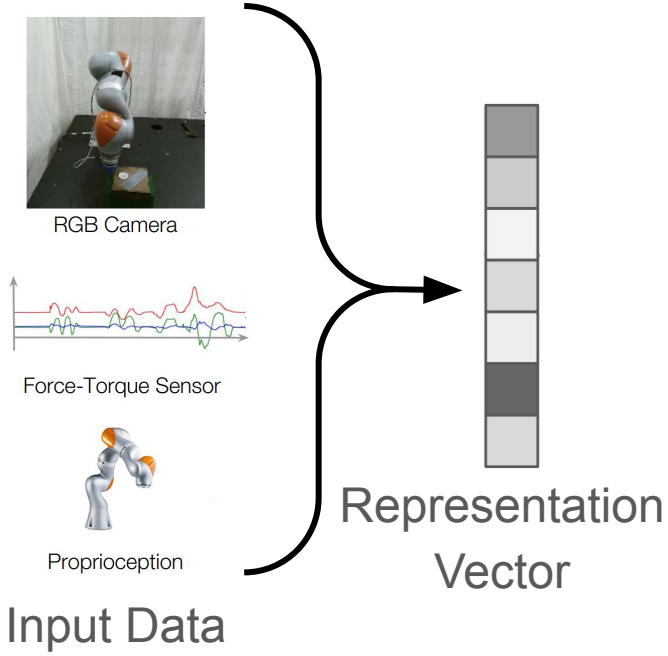
Pre-made Manipulation Graph





Approach - Multi-Modal Representation Model

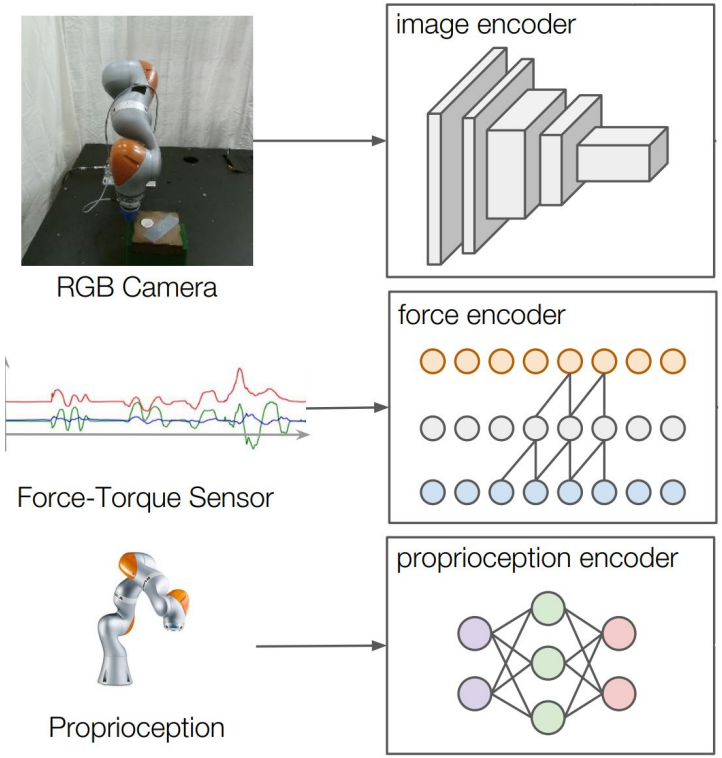
First Objective: Learn representation of high-dimensional data using neural net



Approach - Multi-Modal Representation Model

Challenge: Need to combine three different forms of data into one

Solution: Use separate encoder type for each type of data source



Approach - Multi-Modal Representation Model

Challenges:

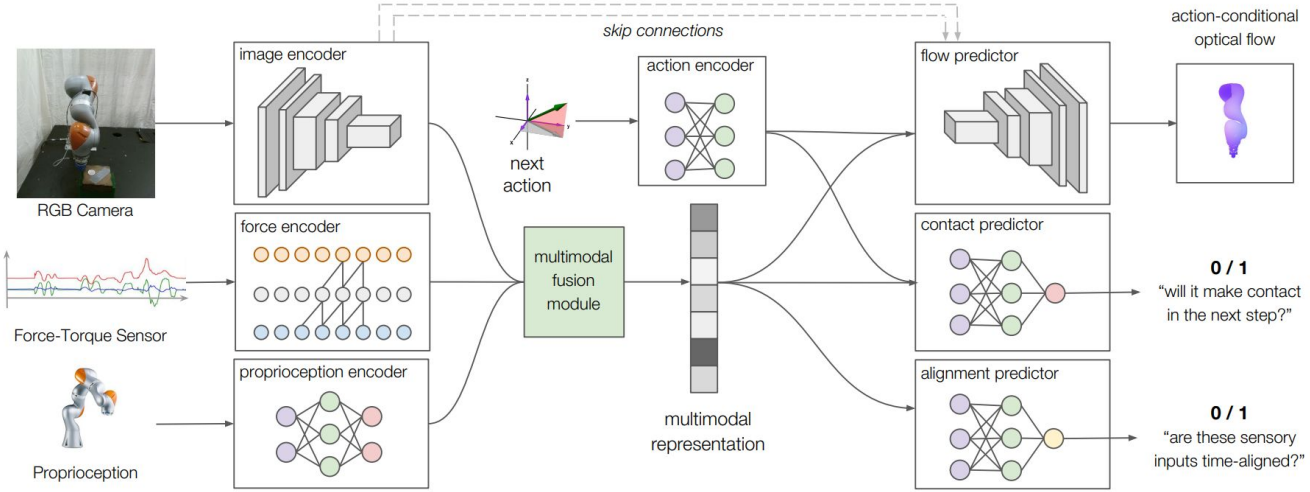
A) Need a lot of training data

B) Want representations to encode action related info

Solutions: Design training objectives that

A) allow for self-supervision

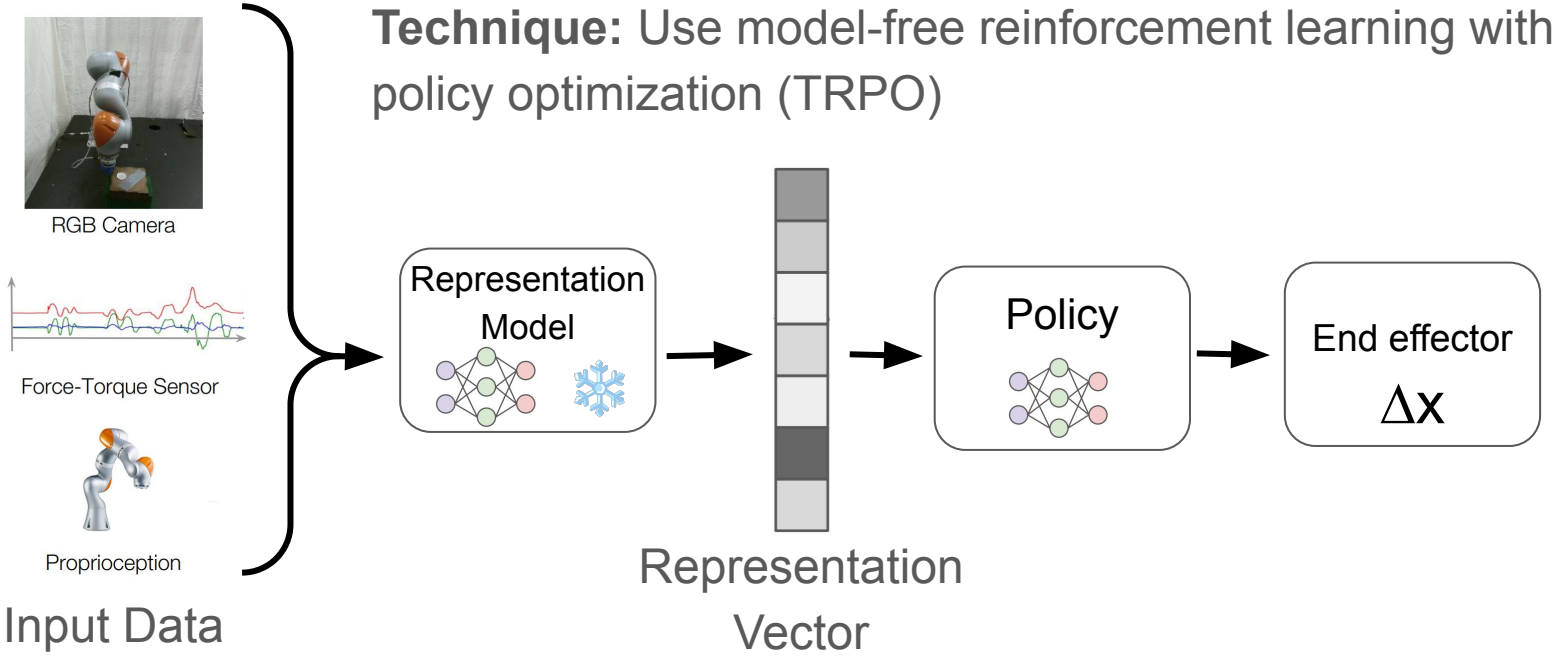
B) use next robot action to make predictions



Approach - Policy Learning & Controller Design

Goal: Train a policy to move the end effector into a desired position

Technique: Use model-free reinforcement learning with trust-region policy optimization (TRPO)

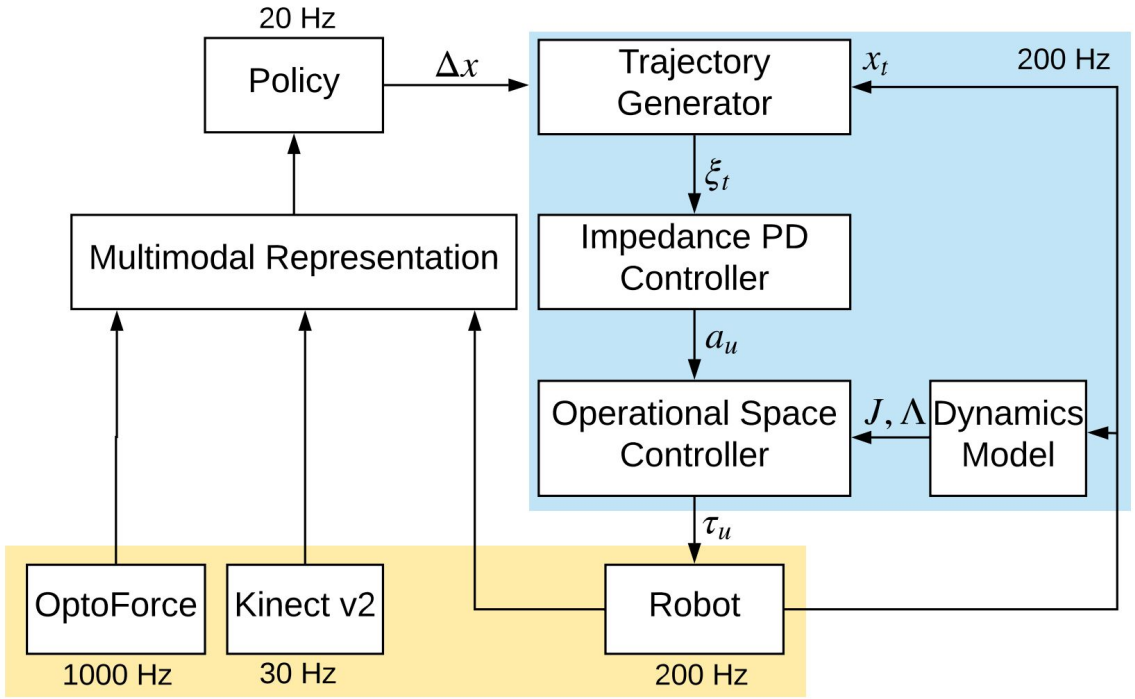


Approach - Policy Learning & Controller Design

Goal: Given 20Hz end-effector Δx, output 200Hz torque commands

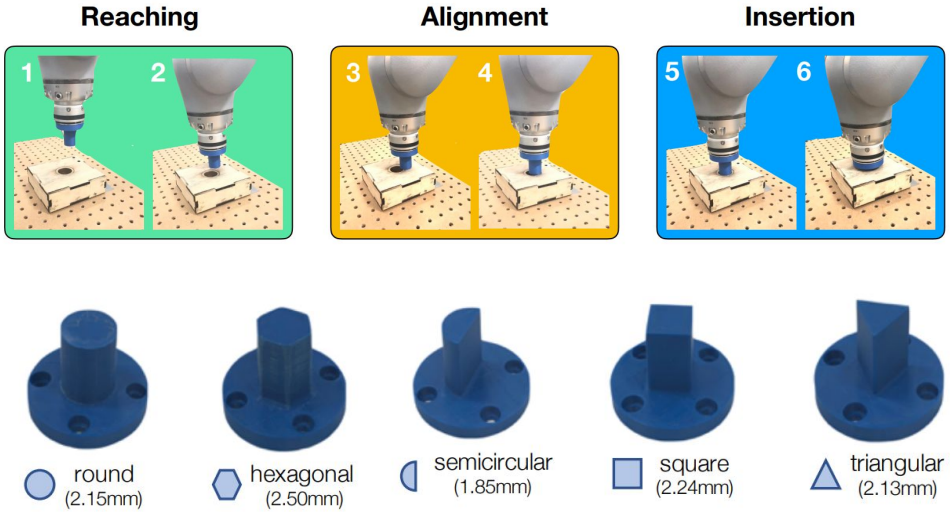
Technique: Use a controller with 3 key pieces

Benefits:
Direct torque control → compliance during contact



Experimental Setup

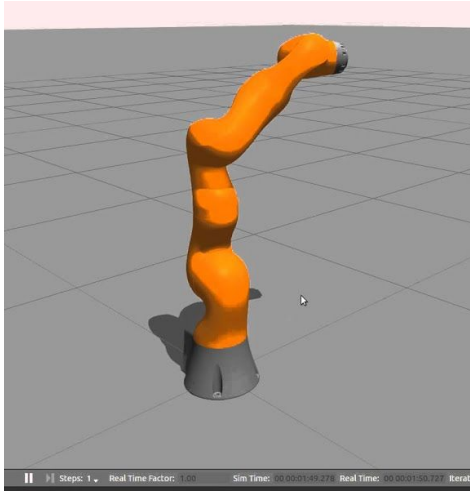
Task: Peg insertion into hole



Environments:

Real Robot

Simulation



Experimental Setup

Staged reward function for training

$$r(\mathbf{s}) = \begin{cases} c_r - \frac{c_r}{2}\left(\tanh \lambda \|\mathbf{s}\| + \tanh \lambda \|\mathbf{s}_{xy}\|\right) & \text{(reaching)} \\ 2 - c_a \|\mathbf{s}_{xy}\|_2 & \text{if } \|\mathbf{s}_{xy}\|_2 \leq \varepsilon_1 \quad \text{(alignment)} \\ 4 - 2\left(\frac{s_z}{h_d - \varepsilon_2}\right) & \text{if } s_z < 0 \quad \text{(insertion)} \\ 10 & \text{if } h_d - |s_z| \leq \varepsilon_2 \quad \text{(completion)}, \end{cases}$$

s - peg's current position

λ, c_r, c_a - constant factors

h_d - height of hole

Experimental Setup - Metrics

Quantitative

Sum of rewards per episode

Qualitative

Task Completion Categories:

Completed Insertion

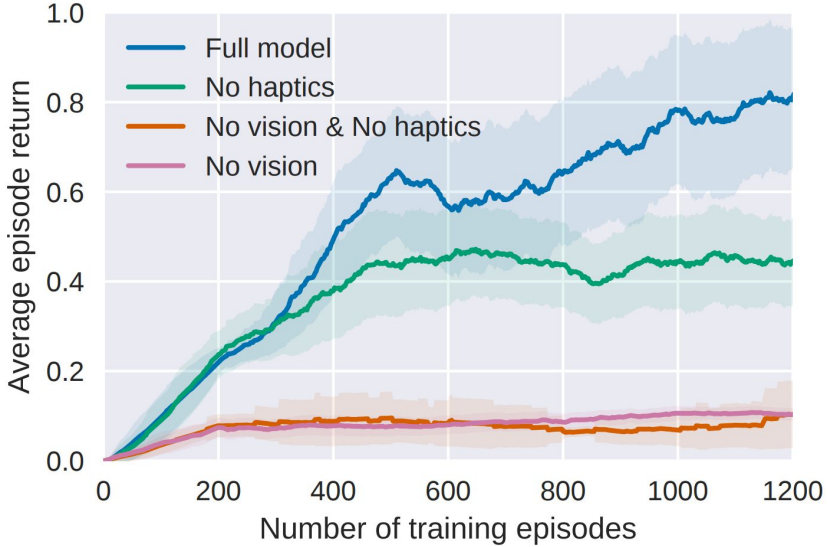
Inserted Into Hole

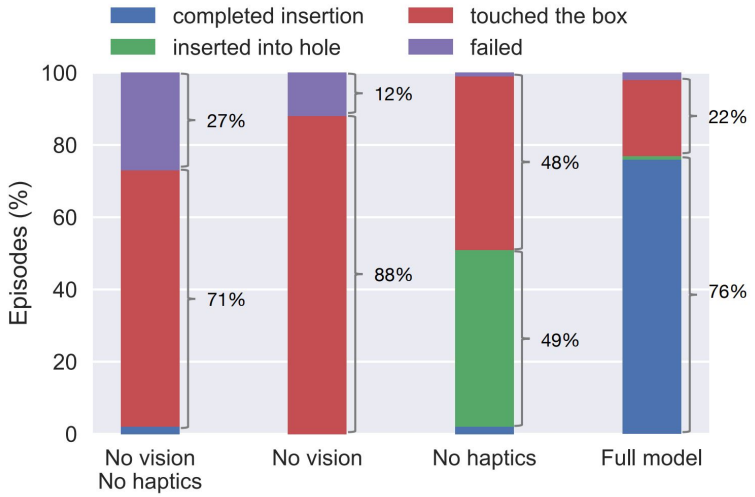
Touched the Box

Failed

Experimental Results - Simulation

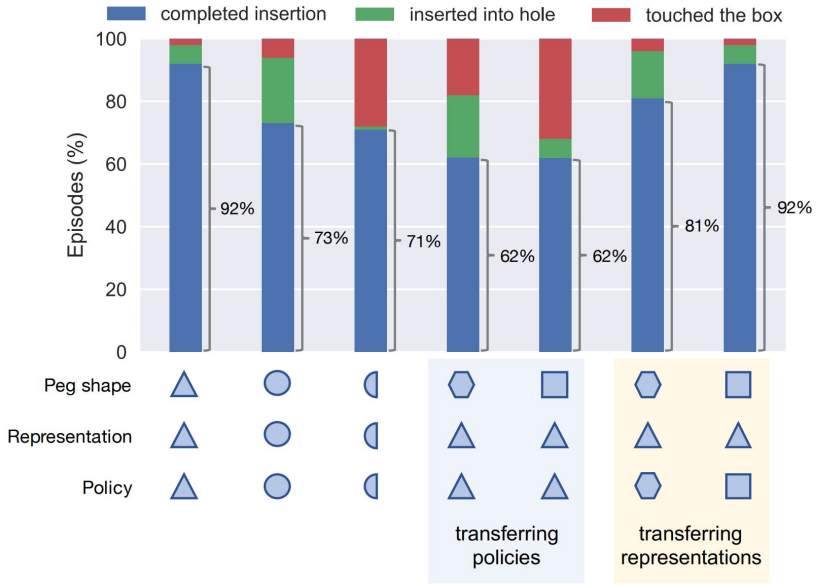
Ablation Study: Determine importance of each sensor modality





Experimental Results - Real Robot

Transfer Learning: How well do the representations and policies transfer between peg shapes?



Results Discussion

- Multi-modal sensory input can greatly improve performance on the peg insertion task
- Learned representations and policies can be transferred between different peg shapes and maintain decent performance
- Policy transfer causes performance to suffer more than representation transfer
  - This is a possible obstacle to generalizing this approach

Critiques / Limitations

- Would have been interesting to compare results of training from scratch on new peg shapes with results of transferring policies and representations

- Possible issues with generalization
  - Policy transfer causes noticeable hit to performance, even on simple task like peg insertion, changing only small part of task (peg shape)

- High training costs
  - To get a robot to perform a task, the robot needs to spend to learning the representation then the policy
  - Limits applications of this technique in real-world scenarios

Future Work

- **Generalized Policies:** Train a single representation and policy on a more varied mix of tasks (i.e. one policy trained on multiple peg shapes). See if this improves policy generalizability

- **Simulation to Real-World Transfer:** Train representations and policies in simulation, then see how well they work in the real world, possibly performing some smaller amount of additional training

- **Extension of current technique:** New tasks, new sensor modalities (e.g. sound, depth)

Extended Readings

Variable Compliance Control for Robotic Peg-in-Hole Assembly: Outlines another approach for using deep RL to perform the peg insertion task, using some techniques not found in this paper

Foundations & Trends in Multimodal Machine Learning: Provides a review of how machine learning can be applied to multimodal inputs

MultiBench: Introduces a set of benchmarks that can be used for assessing multimodal machine learning

Summary

- This paper addresses the problem of training a robot to utilize multiple types of sensory input when performing a task
- Combining visual and haptic input is valuable for improving performance on contact-rich tasks
- The use of deep RL enables learning of more generalizable representations and policies, which was not done in prior work
- **Key Insight:** Multimodal representations can be learned through self-supervised learning, then can be combined with deep RL to achieve high levels of performance on tasks