## Imitation Learning for Skill Transfer in Human-Robot Teleoperation

### Supervisors

R Venkatesha Prasad, Vineet Gokhale, Ashutosh Simha
Embedded and Networked Systems, Faculty of EEMCS

### Background

Teleoperation has gained significant attention in recent years due to its ability to transfer human skills without physical barriers. A robotic arm at the remote environment imitates human operator's actions in order to perform physical tasks, such as pick-and-place, repair machinery, and even medical surgeries, as depicted in Figure 1. This opens a world of fascinating applications, however demanding extensive human intervention. In this project, we aim to enable robotic intelligence by learning task-specific skills from the human via repeated teleoperation.

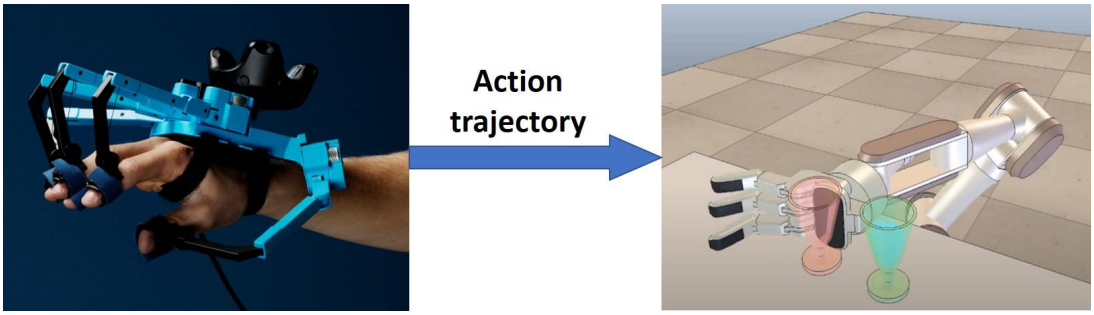


Fig. 1: Human-robot teleoperation with operator relaying action trajectory to the remote robotic arm to perform a task

### Project Aims

The students have to record the operator's arm joint (shoulder, elbow, and wrist) trajectories using kinect and/or a force-feedback glove. These trajectories are then supplied to a simulated robotic arm [1] which reproduces the actions. Initially, the entire trajectory is required by the simulation model. With repeated operations, AI techniques learn the trajectories such that they may be reproduced using minimal inputs, instead of the entire action trajectory, from the operator's side.

**Deliverables**

**Expected**

- Interface output of Kinect/force-feedback glove with simulated robotic arm model such as [1]. Students are free to use a simulation environment of their choice.
- Build an 'action dictionary' which maps each task-trajectory to a small set of parameters. This includes applying AI algorithms in literature for identifying the parameters as well as helping the robotic arm learn and predict operator's actions.

**Optional**

- Perform teleoperation using real network characteristics
- Custom built AI algorithms for task learning.

**Prerequisites**

Knowledge of ML/AI methods; experience in Python and C++ programming.

**Recommended libraries and tools**

Libraries: MS Visual Studio, CoppeliaSim, pandas, numpy, matplotlib

**Provided resources**

- Microsoft Kinect and/or force feedback glove
- Baseline working code for capturing arm trajectory and robot arm simulation model

**References**

[1] https://www.coppeliarobotics.com/