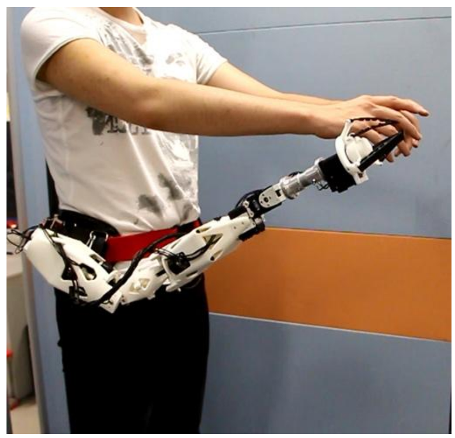
CS486C – Senior Capstone Design in Computer Science

Project Description

Project Title: Reinforcement Learning for a Wearable Robotic Arm – Creating a Sim-to-Real Implementation Framework		
Sponsor Information:	**Zach Lerner, PhD** Assoc. Prof. Mechanical Engineering https://biomech.nau.edu **Prof. Carlo R. da Cunha, Ph.D.** Northern Arizona University - SICCS 1295 S. Knoles Dr., room 115 Flagstaff, AZ 86011 +01(928)523-9339 https://ac.nau.edu/~cc3682	Biomechatronics Laboratory

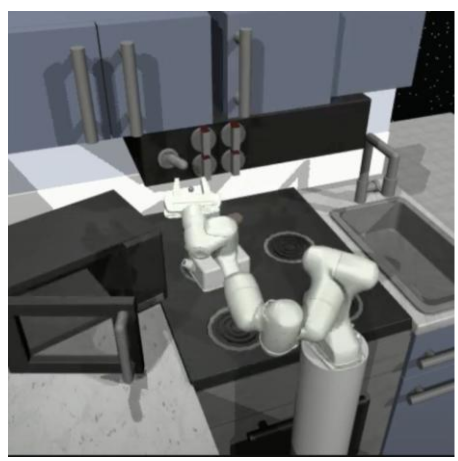
Project Overview:



Background: Stroke is the leading cause of upper limb disability, affecting millions of people worldwide. Survivors typically are unable to adequately move one of their arms. In the Biomechatronics Lab, we are developing a wearable robotic arm to assist with impaired human arm motion.

The problem: The problem is that we do not have a way to control the robotic arm during tasks of daily living in diverse and unpredictable real-world environments.

Solution: This capstone project aims to develop and evaluate a reinforcement learning (RL) framework for controlling a robotic arm to autonomously reach target positions in response to detected objects of interest in its workspace. The project integrates computer vision for object detection with RL-based motion control to enable dynamic and adaptive robotic behavior in unstructured environments.



Traditional robotic arms rely on hardcoded rules or pre-programmed trajectories to reach targets, which limits their adaptability in dynamic settings. There is a need for more flexible and intelligent control systems capable of learning optimal motion strategies from interaction with their environment. Reinforcement learning provides a promising framework for such adaptive control, especially when combined with real-time perception inputs like object detection.

Our team wants to explore **simulation-based reinforcement learning (RL)** as a powerful approach to train robotic systems to learn these behaviors efficiently and safely. By simulating millions of interactions in a high-fidelity physics environment, and then transferring those learned skills to physical hardware, we aim to accelerate development and deployment of our assistive robotic arm.

Capstone Project Specifics and Success Metrics: The goal of this project is to develop a full reinforcement learning (RL) framework for training a robotic arm to assist an impaired human arm during tasks of daily living . This includes implementing the simulation environment, training policy, and domain randomization; stretch goals include sim-to-real transfer and real-time execution on embedded hardware. It is requested that the students implement their solution using RaiSim (https://raisim.com/index.html) as the simulation environment and PyTorch or TensorFlow to train deep RL policies. The objective function should seek to avoid obstacles and collisions, and then find positions that minimize motor power consumption. The robotic system will include joint encoders, and an Oak-D Pro spatial AI camera to provide multimodal inputs. Outputs will include torque or position commands for each joint.

Key Solution Features:

1) Review published research and content on RL control of robotic arms to create a detailed project plan:
    - https://www.nature.com/articles/s41598-025-93175-2
    - https://www.youtube.com/watch?v=z1Lnlw2m8dg (watch entire series)
    - https://www.youtube.com/watch?v=ma6fbvy77Uo (watch entire series)
2) Set up the simulation environment in RaiSim (https://raisim.com)
3) Model our robotic arm in the RaiSim environment, including a human arm that will be supported by the robot.
4) Create different simulation environments (kitchen, bathroom sink, table, etc.)
5) Implement a reinforcement learning policy training using PPO (https://arxiv.org/abs/1707.06347) or SAC algorithms
6) Integration of proprioceptive state input: joint angles, velocities, and tactile data
7) Deployment of policy to the NVIDIA Jetson Orin Nano for real-time control
    a. Integration of the Oak-D Pro robotics camera with object detection algorithm to inform robotic arm motion

Knowledge, skills, and expertise required for this project:

- Python and/or C++ programming - required
- Deep learning frameworks (PyTorch or TensorFlow) - desired
- Basic reinforcement learning (PPO, SAC) - desired
- Physics simulation (RaiSim or MuJoCo) - desired

Equipment Requirements:

- Modern laptop/desktop computers
- Access to Monsoon
- Freely available software

Software and other Deliverables:

- A report detailing the design and implementation of sufficient detail to allow anyone to replicate the work
- Complete, professionally-documented codebase in GitHub

- Development tracked using Sourcetree
- Trained policy model and simulation assets
- Videos of successful simulation training