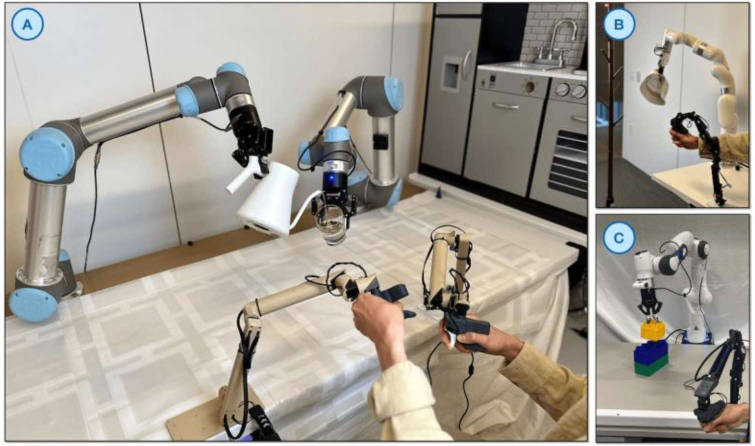
Construction of a low-cost dual arm teleoperation interface

Problem description: Recent trends in Deep Learning and Robotics require collecting vast amounts of robot motion demonstrations, for the robots to learn the manipulation policies. Modern teleoperation devices typically cost thousands of Euros; however, a recent line of works replace these devices with cheap, devices made of simple motors, aluminum extrusions, and 3d printed parts. The goal of this project is to develop such a device that would allow for teleoperating a dual arm robot manipulator.

Tasks:

1. Familiarize with GELLO, a popular hardware platform for low-cost robot teleoperation.
2. Desing and build a teleoperation device based on the GELLO Project for the dual-arm robot setup.
3. Evaluate the teleoperation performance by collecting demonstrations for a simple task (pick, pouring, gear insertion, etc.), learning the respective skills (DMP, GMM/GMR, etc.), and executing it simulation and real robot system.



GELLO webpage: https://wuphilipp.github.io/gello_site/

Supervisor: Shail Jadav; shail.jadav@tuwien.ac.at

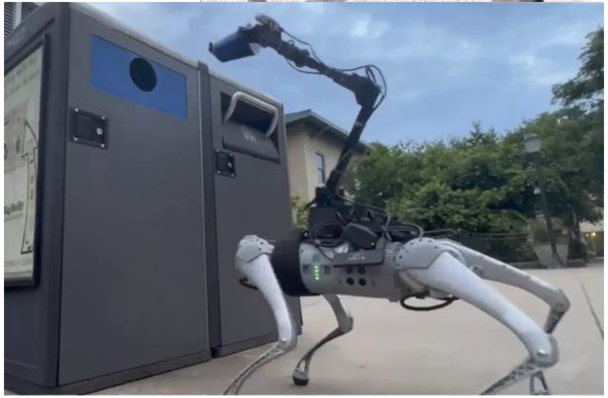
Number of students: 1

Manipulation learning for assistive quadruped robots

Problem description: Quadruped robots have become increasingly affordable and robust, making them more accessible to regular customers. For these robots to assist humans effectively, they must possess the ability to manipulate objects in their environment. One common solution involves equipping quadruped robots with robotic manipulator arms, enabling them to interact with objects in the environment. However, teaching these robots manipulation skills remains an open challenge. To address this, we would like to use Learning from Demonstration (LfD) approaches like Dynamic Movement Primitives (DMPs), Gaussian Mixture Models (GMMs), or Kernelized Movement Primitives (KMPs).

Tasks:

1. Recording of manipulation skills through kinesthetic teaching or teleoperation
2. Implementation of a state-of-the-art LfD approach for encoding manipulation skills
3. Execution of manipulation skills on the Unitree Go2 robot
4. Validation of the approach in 3 tasks
   - pick and place
   - opening and closing of shelfs or drawers
   - pressing buttons or switches



Supervisor: Johannes Heidersberger, johannes.heidersberger@tuwien.ac.at

Number of students: 1

Universal Manipulation Interface Design

Problem description: Universal Manipulation Interface (UMI) is an open-source handheld robot teaching tool developed by Stanford. It is used to teach robot manipulation skills without the need of a teleoperated robot, greatly increasing the speed of collecting data. The project goal is to design and implement a similar teaching device and evaluate its performance in learning robot manipulation skills.

Tasks:

1. Improve upon the current design on the teaching device to optimize the weight.
2. Print and build a prototype of the revised teaching device.
3. Collect a dataset of skill demonstrations for manipulation tasks like gear insertions, peg insertion, or cable harnessing (only one task; provided ones are just examples).
4. Learn a manipulation policy with existing state-of-the-art encoding frameworks.
5. Execute learn policies on a Franka Emika Panda robot.

UMI webpage: https://umi-gripper.github.io/

Supervisor: Shail Jadav; shail.jadav@tuwien.ac.at

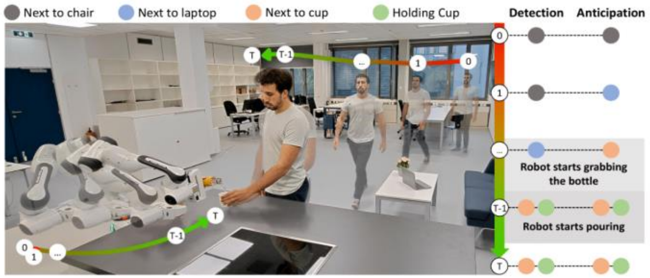
Number of students: 1

Let me help you! Intention Guided Task Planning for Physical Collaborative Assembly Tasks

Problem description: The goal of this project is to develop a framework for human-robot collaboration in assembling IKEA furniture. This involves integrating cutting-edge methods for human intention detection and ensuring safe physical interactions, where the task's autonomy is shared between the robot and the human. Behavior trees will be used to guide the execution of the assembly process. Successful completion of the project will provide valuable insights for factory automation, particularly in scenarios where humans and robots must work together efficiently.

Tasks:

1. **Create a Behavior Tree for IKEA Assembly:** Develop a modular plan that outlines each step of the assembly process, breaking it into tasks like fetching parts, using tools, and combining components, while managing contingencies like re-attempts.
2. **Integrate Human Intention Detection and Physical Human-Robot Collaboration:** Seamlessly combine the intention detection model [1] with our framework for physical human robot interaction [2], ensuring the robot can assist humans safely and efficiently.
3. **Conducting Experiments:** Validate the effectiveness of the integrated system in real-world IKEA assembly scenarios and refine the system based on experimental findings.



Supervisor: Daniel Sliwowski; daniel.sliwowski@tuwien.ac.at

Number of students: 1

[1] https://evm7.github.io/HOI4ABOT_page/
[2] https://shailjadav.github.io/SALADS/

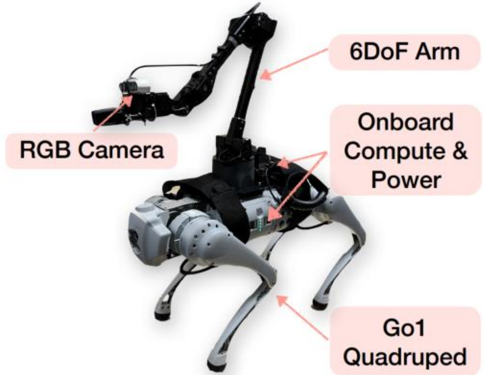
Whole-Body Manipulation through Reinforcement Learning

Problem description: Attaching a robot arm to quadrupeds can significantly increase their applicability to everyday tasks, such as opening a door, or assist human for some tasks. To manage different tasks. the robot may need to coordinate its arm and legs, for example, it might need to raise its legs to reach objects or bend its front legs to approach front objects, or it has to change its legs to produce enough torques to open a door. In this project, students will apply deep reinforcement learning (DRL) methods to learn a whole-body control policy to control both legs and the arm to different tasks.

Task: The robot learns to coordinate its legs and arms to extand its reach zone or produce enough force for the arm to manage different daily tasks. During training the objects will be placed at random positions (e.g., at different directions of the robot, on a floor, on a table, etc.). The robot should fetch them successfully. To achieve this, student will:

- research literatures on DRL for robot manipulation.
- implementation of DRL methods on a unitree robot with an arm.
- train your controller in a simulation environment.
- (optional) implement your method to the real robot.

Supervisor: Yashuai Yan (yashuai.yan@tuwien.ac.at)
Number of students: 1

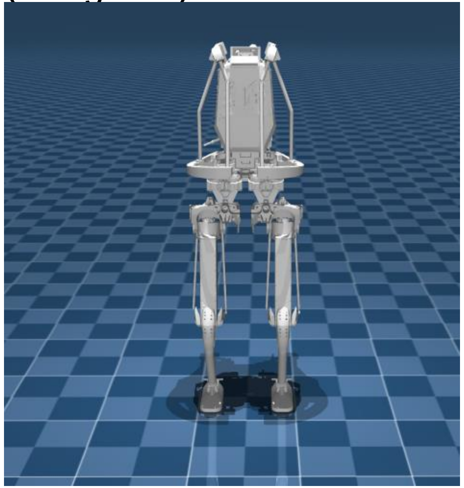


Hardware setup (provided in our lab)

[1] Z. Fu, X. Cheng, and D. Pathak. Deep whole-body control: learning a unified policy for manipulation and locomotion. In Conference on Robot Learning, 2022

Bipedal Robot Locomotion with Reinforcement Learning

Problem description: In this project, student will apply deep reinforcement learning (DRL) to develop a walking controller for a bipedal robot (Kangaroo) with two 12 DoFs legs.



Kangaroo Robot in Simulator

Task: Apply DRL to train a walking control policy for the Kangaroo to enable control it with velocity command. To achieve this, the following steps are recommended:

- Create a simulation environment for the Kangaroo robot using MujocoX/IsaacGym. .
- Design and implement a training pipeline for the control system in simulation.
- Apply DRL algorithms to learn a control policy.
- Train a control policy that enables the Kangaroo robot to follow velocity commands.

Supervisor: Yashuai Yan (yashuai.yan@tuwien.ac.at)
Number of students: 1

Robot manipulation with Reinforcement Learning

Problem description: In this project, students will employ deep reinforcement learning (DRL) to develop robotic manipulation skills, focusing on tasks such as pick-and-place operations and connector insertion, which are critical for assembly processes. The flexibility in choosing a specific reinforcement learning algorithm allows to explore various DRL approaches, such as value-based methods (e.g., DQN), policy-based methods (e.g., PPO), or actor-critic architectures. The primary objective is to design a control strategy that enables the robot to learn these skills autonomously through interaction with its environment, improving its performance over time via trial and error.

Supervisor: Jedrzej Orbik (jedrzej.orbik@tuwien.ac.at)
Number of students: 1

Research robotic arm for sim training

Task: Apply selected DRL method to train a set of control policies for the robotic manipulator. To achieve this, the following steps are recommended:

- Create a simulation environment for the robotic arm robot using MujocoXLA/IsaacLab
- Design and implement a training pipeline for the control system in simulation.
- Apply DRL algorithms to learn a set of policies – one per task
- (optional) evaluate the policy learning in the real-world

Creating low-cost robotic HW & SW for manipulation

Problem description: In this project, students will design and construct a low-cost robotic arm utilizing off-the-shelf actuators and components.

Exemplary low-cost robotic arm

Task: Create the hardware and software for the low-cost robotic arm

- Start from scratch on the design and assembly of the robotic arm
- Write the software stack for the robotic arm control
- Demonstrate the trajectory tracking using the developed hardware
- Examples for inspiration: https://github.com/jess-moss/koch-v1-1 or https://emanual.robotis.com/docs/en/platform/openmanipulator_x/overview/

Supervisor: Jedrzej Orbik (jedrzej.orbik@tuwien.ac.at)
Shail Jadav (shail.jadav@tuwien.ac.at)
Number of students: 1