



Assignment

Coding challenge - Teleoperation Challenge

Deadline:

You will have 7 days to complete the challenge.

Tasks:

1. Set Up the Environment

Develop the project as a **ROS2 Humble** workspace.

Use a computer equipped with a camera.

2. Robot Arm Visualization

Set up a pair of dual arms (both 6DOF and equipped with two-finger grippers) in a visualization platform such as **RViz** (not a simulator).

Implement the corresponding motion planning (with self-collision avoidance) for the dual arms.

3. Message Publishing

Publish the current joint angle values from the visualization platform as ROS messages.

4. Teleoperation Node

Create a **teleoperation ROS node** that allows the operator to control the end-effector movement with their hands and/or body. The node should :

Captures and tracks the motion (hand gesture, posture, ...) of the operator using images from the camera.

Generates and publishes corresponding end-effector target poses of the dual arms.

Allows control of both the movement of the end-effectors and the open-close action of the grippers. Important: Since we do not expect you to have a depth camera, you can just control the end-effector to move within a plane and rotate along one axis (2D Movement + 1D Rotation).

1. Version Control

Your project must include a version control system (e.g., Git). Please also include a README file describing how to setup/install your solution and run in a new machine (extra libraries/packages to be installed, CAD files/meshes to be downloaded, etc). The version control could be just

local, it's not mandatory to make it available online. If you decide to upload it online (on GitHub, GitLab, etc), please make the repository private and ask Sereact who should be invited to your repo.

2. Video Demonstration

Provide a video showcasing your solution in action.

3. References to Open-Source Code

You may refer to open-source repositories, but ensure to acknowledge them in the README and fully understand their implementation.

Attention:

- Perfect precision in (hand) pose recognition and tracking is **not required**, but your solution should include measures to handle noisy results.
- Design the system with the real-world scenario in mind: the real robot arms will subscribe to the published messages and reach the joint values within milliseconds.
- Ensure that the solution can be safely operated by only **one human operator**.
- The reachability of the human body is limited.
- Try to make the movement swift, stable and smooth

Note:

Feel free to reach out through Mail in case you have any questions.

Submission

Files

 **Upload File(s)**

or drag and drop here

Notes

Write here...

Submit Assignment