Notes on Thesis Project

Salvatore De Luca
git/salvatore.deluca

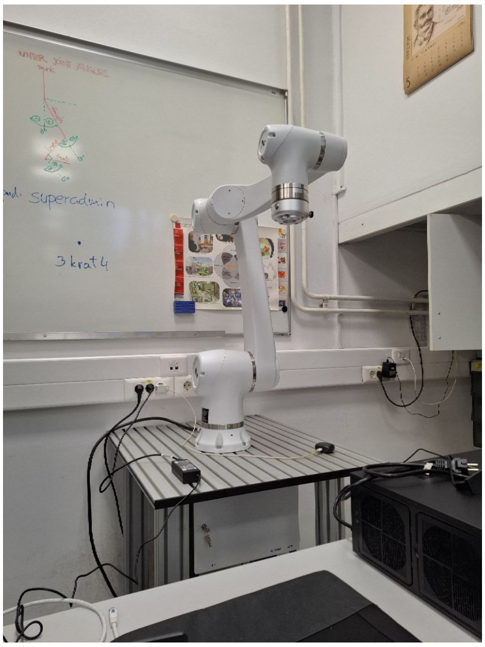
November 2025

1 Current Progress

The initial phase of the project focused on hardware setup and software architecture definition. The activities carried out so far can be summarized as follows:

Initial Setup and API Testing

- **Hardware Assembly:** The robot was assembled and commissioned.
- **Preliminary Testing:** Operation was first verified using the Teach Pendant, followed by an evaluation of the available interfaces (C++ API, Python API).
- **Identified Limitations:** The native libraries allowed only for simple movements. Crucially, they lacked support for trajectory generation that accounts for collision constraints.



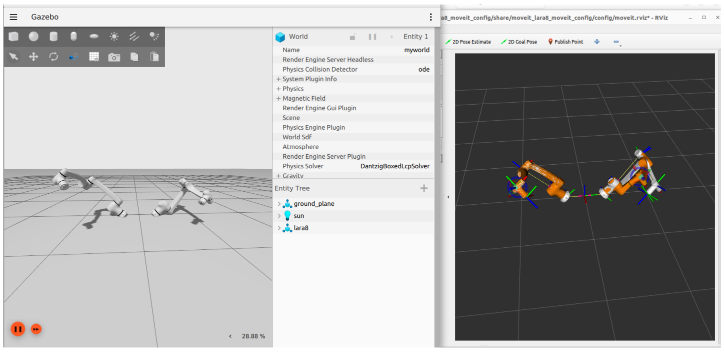
Transition to ROS2

Due to the limitations of the proprietary NEURA software, the decision was made to structure the interface using **ROS2**.

- **Motivation:** To overcome the inability of the native API to handle complex constraints and to enable advanced planning.
- **MoveIt! Configuration:** A MoveIt! package was configured for the dual-robot system.
- **Validation:** The MoveIt C++ API was successfully tested both in simulation (RViz GUI) and on real hardware (currently on a single robot).

Visual Servoing Preparation

- **Simulation Environment:** To support planned visual servoing tasks, a camera sensor was integrated into the simulation environment, and a simulation instance was launched within **Gazebo**.

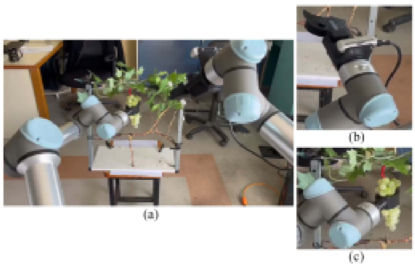


2 Methodology

The methodology is grounded in an investigation of previous results concerning dual-arm robots in agricultural contexts. The following key studies were analyzed:

Case Study 1: Bimanual Grape Manipulation[1]

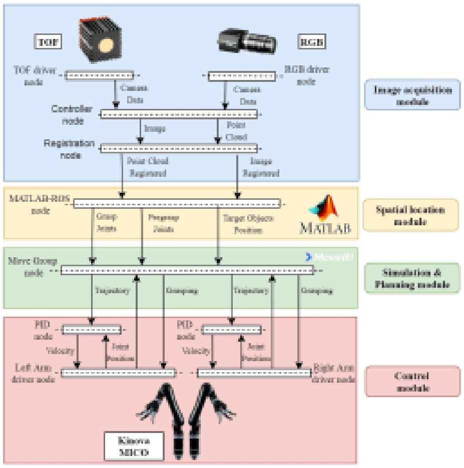
Focus: Human-Inspired Robotic Harvesting



- **Role Division:** One robot handles grasping; the other (equipped with a camera) handles observation and cutting.
- **Process:** The camera arm detects the fruit and maintains it in the Field of View (FoV). The grasping arm secures the fruit, allowing the camera arm to cut it.
- **Control Objectives:**
  1. **Camera Arm:** Reaching, centering, and unveiling the fruit.
  2. **Grasping Arm:** Force/position parallel control to reach the target and regulate grasping force (preventing damage).
  3. **Correction:** A velocity-resolved control law to keep the fruit in the camera's FoV.

Case Study 2: Robotic Aubergine Harvesting[2]

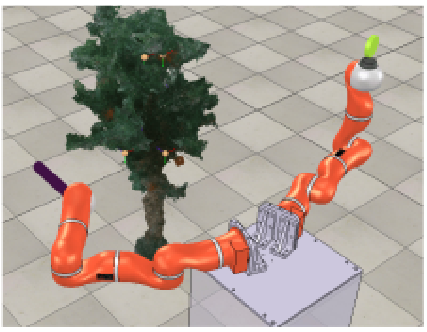
Focus: Task Allocation and Efficiency



- **Detection:** Uses a Support Vector Machine (SVM) pixel-based classifier.
- **Task Planning:** An algorithm efficiently distributes grasping tasks between arms.
- **Obstacle Avoidance:** Requires a specific image processing step to detect obstacles.
- **Execution:** MoveIt generates waypoints which serve as input for the robot's PID controllers.

Case Study 3: Vision-Based Dual Arm Fruit Harvesting[3]

Focus: Hierarchical Control and Optimization



- **Perception Pipeline:**

  - Image acquisition and 3D localization of the target fruit (with ID).
  - Determination of the cutting point (pistil distance).
  - Tree localization (segmentation and filtering of branches/trunk).

- **Task Definition:**
  - **Absolute Task:** Cutting the fruit.
  - **Relative Task:** Grasping the fruit (planned in the cutting arm's frame).

- **Control Strategy: Hierarchical Quadratic Programming (HQP)** solves priority tasks ensuring a decoupled solution.

- **Collision Avoidance:** Modeled using variable-radius cylinders. A speed damping mechanism slows down bodies when distance $< d_s$.

Note on Reinforcement Learning

Alternative solutions using Reinforcement Learning (RL) were considered. RL would require redesigning the entire setup, rendering the current MoveIt implementation (which solves path planning with a priori collision avoidance) redundant.

3 Future Development

The next steps focus on completing the hardware integration and defining the final control strategy:

- **Hardware Completion:** Assembly of the second robot.

- **Dual-Robot Integration:**
  - Test synchronous command transmission via ROS2.
  - Validate MoveIt integration for the complete dual-arm system.

- **Control Strategy Decision:**
  - *Option A:* Continue sending MoveIt-generated waypoints.
  - *Option B:* Transition to direct velocity control. This would involve using `ros2_control` or exploring **MoveIt Servo** (Realtime Arm Servoing), although the documentation is poorer.

Bibliography

[1] S. Stavridis, L. Droukas, and Z. Doulgeri, "Bimanual Grape Manipulation for Human-Inspired Robotic Harvesting," *IEEE/ASME Transactions on Mechatronics*, vol. 30, pp. 2722–2732, 4 2025. DOI: `10.1109/TMECH.2024.3459479`.

[2] D. Sepúlveda, R. Fernández, E. Navas, M. Armada, and P. González-de-Santos, "Robotic Aubergine Harvesting Using Dual-Arm Manipulation," *IEEE Access*, vol. 8, pp. 121 889–121 904, 2020. DOI: `10.1109/ACCESS.2020.3006919`.

[3] E. Gursoy, B. Navarro, A. Cosgun, D. Kulić, and A. Cherubini, "Towards vision-based dual arm robotic fruit harvesting," in *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, IEEE, 2023, pp. 1117–1122. DOI: `10.1109/CASE56687.2023.10260651`.

[4] *Ros2 guide for MAiRA/LARA robots*, Document version: v1.0; Marzo 2025, NEURA Robotics GmbH, 2025.

[5] *Neura cpp api*, version 1.1, Release 1.1; Data: 17 Maggio 2025, Neura Robotics GmbH, May 2025.

[6] *Neurapy*, version v4.22.4, Release v4.22.4; Data: 17 Maggio 2025, Neura Robotics GmbH, May 2025.

[7] *Lightweight agile robotic assistant software manual*, 100002000002, version v5.0.0, NEURA Robotics GmbH, Jun. 2025.

Sitography

[8] M. C. (GitHub). "Dual_arm_panda_moveit_config demo launch file," Accessed: Nov. 14, 2025. [Online]. Available: `https://github.com/moveit/moveit_resources/blob/ros2/dual_arm_panda_moveit_config/launch/demo.launch.py`.

[9] P. Robotics. "Moveit documentation (main index)," Accessed: Nov. 14, 2025. [Online]. Available: `https://moveit.picknik.ai/main/index.html`.

[10] ROS Developers. "Namespace moveit," Accessed: Nov. 14, 2025. [Online]. Available: `https://docs.ros.org/en/lunar/api/moveit_ros_planning_interface/html/namespacemoveit.html`.