Relative Proximity Operations

Kaylee Hudson, Harry Furey-Soper

BLUE ORIGIN

WILLIAM E. BOEING
DEPARTMENT OF AERONAUTICS & ASTRONAUTICS
UNIVERSITY of WASHINGTON

Washington Space Grant Consortium
NASA

Can a Camera Fly a Spaceship?

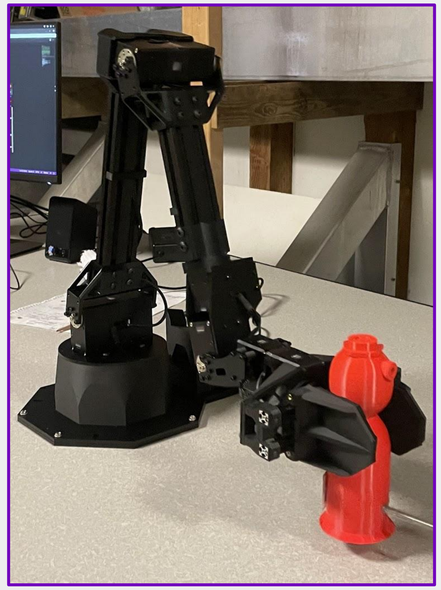
A recent trend in robotics research is the use of relative motion tracking cameras to control the movement of multiple robots working in a group. This project aims to push the boundaries of such technologies with an application in the astronautics field: **using cameras to guide two spaceships to dock with each other autonomously.**

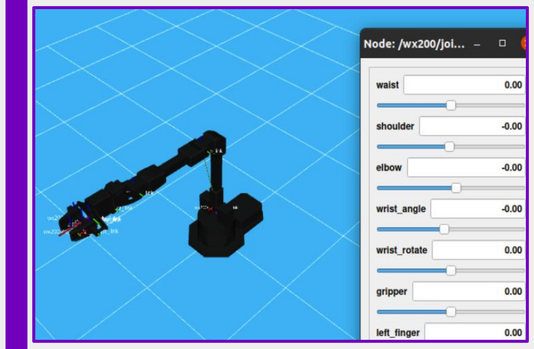
The Test Models

To begin this project, 3D printed spacecraft models were equipped with visual trackers and camera mounts. The D415 camera was chosen for it's **ability to sense depth**, which is important for tracking motion in 3 dimensions. The visual trackers are simple 2D barcodes known as **AprilTags**. Model shown is a portion of the International Space Station.

Motion Testing

Two Trossen Robotics ViperX 300s robotic manipulator arms are used to move the test models. They can be programmed to **simulate scaled-down orbital trajectories**. They can also be operated in tandem with each other, to simulate **docking maneuvers between two vessels** carried by the two separate arms.





Virtual Testing

Python scripts used to control the arms are tested using a **virtual robotics simulator**.

Programming with ROS

ROS, or **Robot Operating System**, is a collection of software libraries which is commonly used in the robotics industry. Our current task is to determine which combination of ROS version, compatible interfacing softwares, and linux operating system is **best suited to achieve the long-term goals** of the project using the ViperX 300s robotic arms. So far, we have been investigating the following:

- **Linux OS:** Ubuntu 18.04, 20.04, and 22.04 on several desktop computers, dual-boot machines, and virtual machines
- **ROS Versions:** ROS Noetic, ROS Melodic, and ROS2 Galactic
- **Simulators:** RViz and Gazebo
- **Motion Planning Packages:** MoveIt, Python API

Current Goals

- Have arms individually carry models through scaled-down orbital trajectories.
- Achieve reliable collision avoidance with both arms operating in tandem.

Summary: What We've Done So Far

- **Fabrication+Setup:** 3D printed spacecraft models, mounted cameras and AprilTags, built testbed for robotic arms.
- **Visual Processing:** MATLAB was used to analyze photographs of AprilTags under varying conditions. It was found that the accuracy of the tags decreased with distance and orientation angle relative to the camera, and increased with tag size and curvature radius.
- **Software:** Established MoveIt as our most efficient motion planning tool. Ruled out virtual machine use for physical arm control, but continuing use in virtual testing. Decided on RViz over Gazebo for simulations. Narrowed down to ROS Noetic and Ubuntu 20.04 for physical testing.

Future Work

- Study the effects of AprilTag curvature and orientation more closely.
- Use the cameras to guide the motion of individual robotic arms with respect to stationary AprilTags.
- Operate arms in tandem using cameras to autonomously guide docking maneuvers.

Acknowledgements

The work in this project was supported in part by funding from the Washington NASA Space Grant Consortium and in part by a contract with Blue Origin.