Virtual Reality (VR) Interface for Robot Teleoperation and Environment Visualisation

Supervisors:
Dr Alan Millard
Dr Alena Denisova

Author: **Alistair Foggin**

Computer Science
University of York

1. Motivation

Reasons for a Virtual Reality teleoperation interface

- Enable immersive teleoperation of robots in challenging environments
- Reduce mental load of understanding sensor data
- Implement an interface that matches human mental models

2. Design/Development Process

Phases and steps followed in developing the interface

Data Collection

- Surveyed 13 robotics researchers on their work needs
- Extracted survey results into key points and themes
- Analysed results to identify related responses, tools, capabilities, and challenges using an affinity diagram
- Created personas and scenarios from responses

Design

- Researched capabilities of tools mentioned in the survey
- Identified user and system requirements based on survey results
- Followed Virtual Reality usability guidelines
- Brainstormed and sketched possible interface layouts

Development

- Started prototyping the interface
- Asked volunteers to test the prototype
- Iterated on prototype based on feedback

3. Survey Results

Summary of responses from the survey

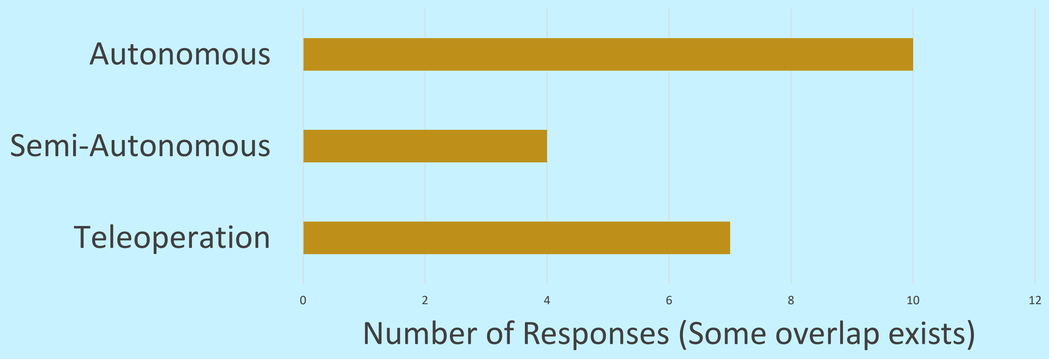
Usability Requirements

Consistent | Easy to use | Responsive

Pre-existing Frustrations and Difficulties

Internet and Latency | Interfacing with GUI Services | Complicated Build Tools

Levels of Autonomy

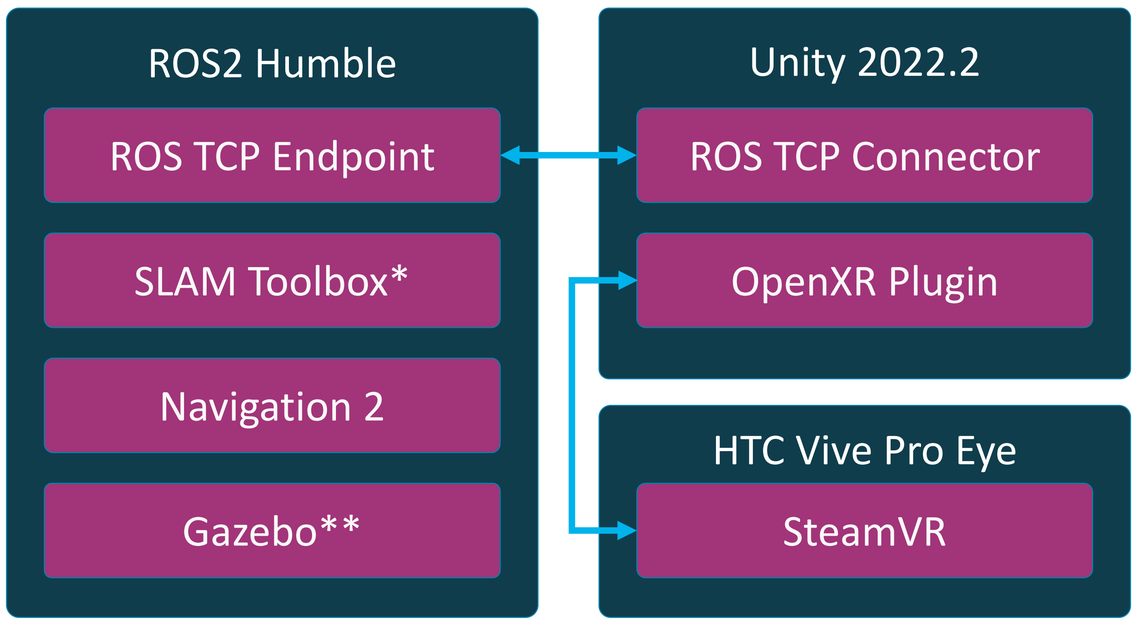


Existing Tools and Sensors

Visualisation and Control	Simulation	Sensors
• RViz	• Gazebo	• LiDAR
• Foxglove	• CoppeliaSim	• IMU
• Unity	• Argos	• Camera
• Formant	• Unity	• Stereo Camera
	• MATLAB	• IR

4. System Overview

Collection of components required to create the interface in Unity



*SLAM Toolbox can be swapped out for Cartographer
**Gazebo is only necessary when simulating the Turtlebot3

5. Interface Implementation

Summary of the features in the interface

Virtual Environment

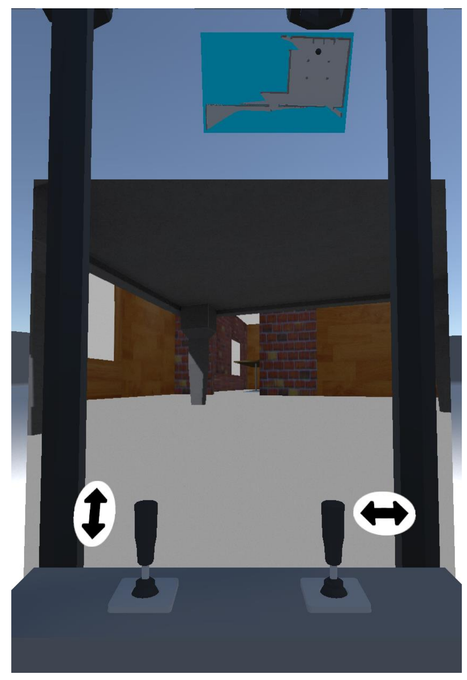
- Virtual Cockpit
- Teleoperation Dashboard
- UI Control Panels

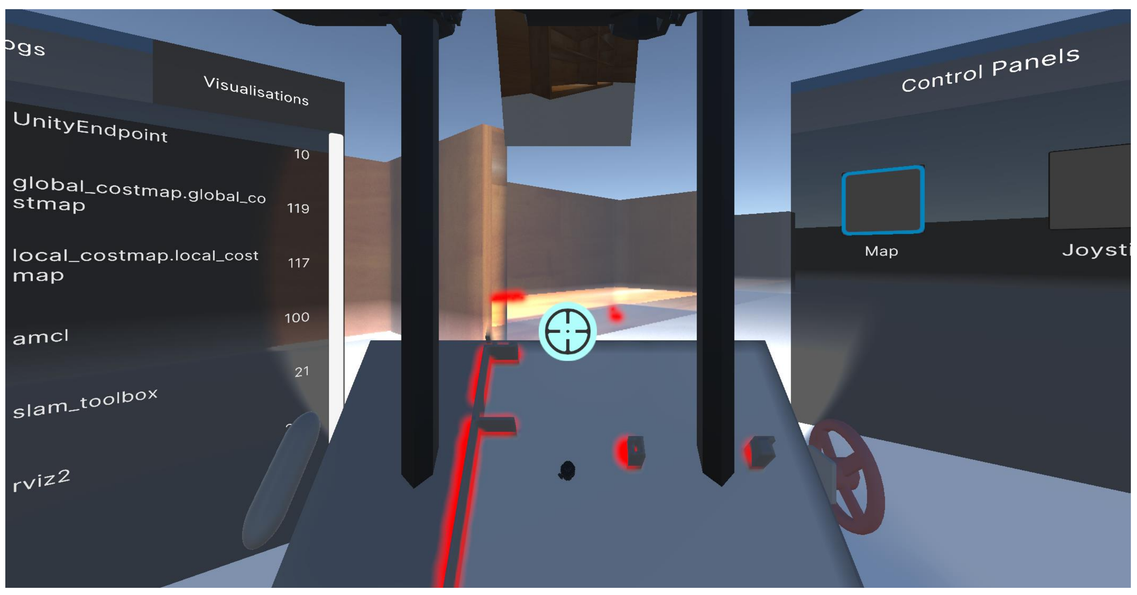
Visualisation

- 3D SLAM Map
- LiDAR
- Camera

Teleoperation

- Point and Click Navigation
- Joystick Directional Controls





6. Closing Evaluation

Details about challenges encountered and work to be expanded upon

Challenges in Development

- Network Bandwidth
- Limitations of Unity-ROS Integration
- Ensuring consistent high framerate in VR

Potential Future Work

- Improve map gesture controls
- Add more sensor visualisations
- Dynamically integrate with different robots using URDF
- Switch between multiple robots at runtime

alistair.foggin@york.ac.uk
https://uk.linkedin.com/in/alistair-foggin
https://github.com/alistairfoggin/YorRobotsVR

Funded by
YorRobots Venables Internship