A Low-Cost Open-Source Visual Odometry Solution for Coastal ROV

SEATTLE AQUARIUM

INTRODUCTION

- Remotely operated vehicles (ROVs) are often used for surveying shallow subtidal regions
- BlueROV2 lacks the ability to measure its velocity and altitude, assisting in navigation
- Doppler velocity logs (DVL) are a solution; however, they are too expensive
- Our project aims to integrate a visual odometry system with a BlueROV2 and make real time measurements

REQUIREMENTS

- Achieve no more than 1% error in velocity measurements compared to ground truth
- Maintain accurate depth measurement within 5% of true value
- Process and deliver data at minimum 5Hz, optimal 10Hz
- Function reliably in variable underwater conditions with limited visibility
- Real-time computation on Jetson Nano with maximum 300ms latency from image capture to velocity output
- Interface with ROV's Raspberry Pi via Ethernet/WiFi for data transmission

SYSTEM ARCHITECTURE

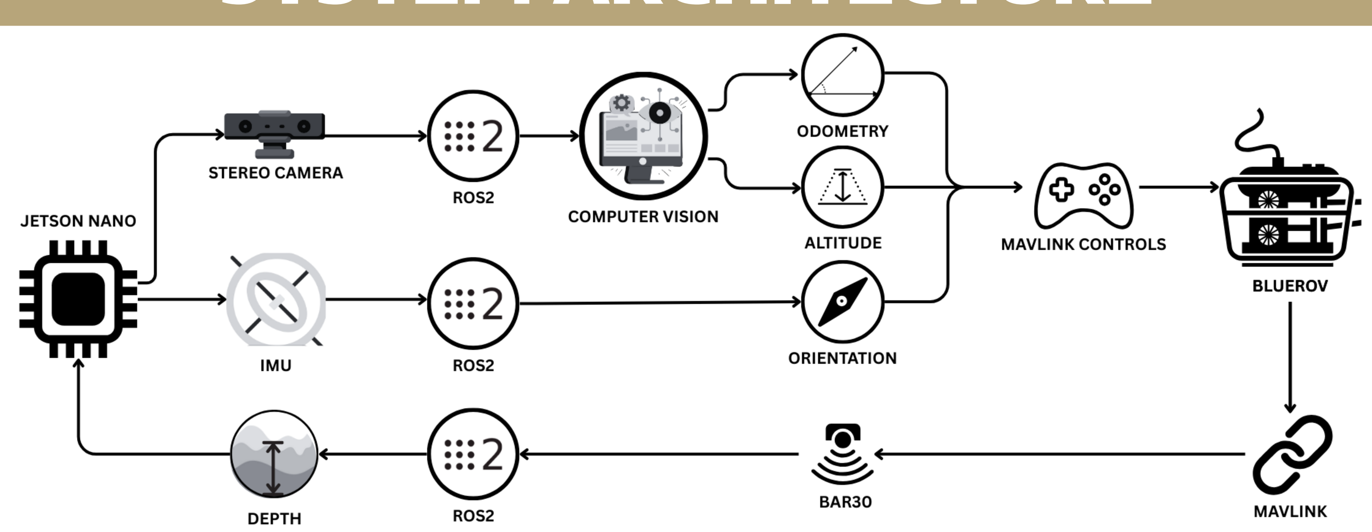

Figure 1. System Architecture

- Our system consists of software and hardware components, where software system is implemented on a Jetson Nano, communicating with the hardware components via Robot Operating System 2 (ROS2)
- After calibration for lens distortion and object sizing, stereo camera and IMU drivers publish data to odometry wrapper, which is then sent to ORB-SLAM3 to compute for pose, orientation, and mapping
- After the computation, ORB-SLAM3 sends back the pose and orientation data to the odometry wrapper as the source to compute odometry and altitude data
- Odometry, altitude, and orientation data from IMU are sent through MAVLINK to the EKF on Raspberry Pi board on BlueROV2
- Finally, BlueROV2 can also transmit depth value of BAR30 to the Jetson Nano to verify the accuracy in known and controlled environments

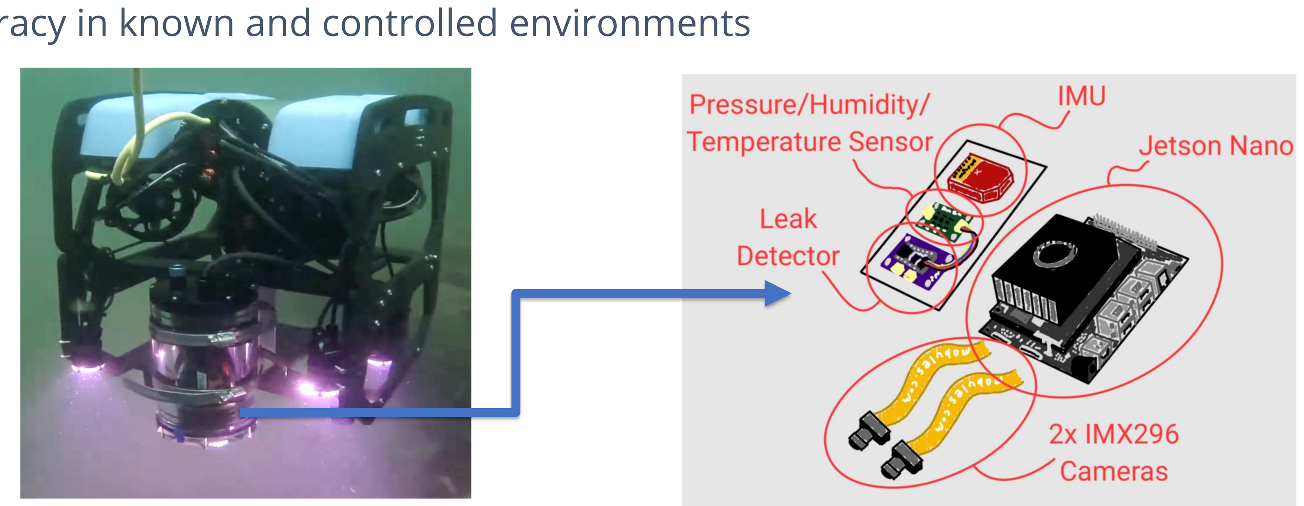

Figure 2. Computational hardware of our system (right) is enclosed in the capsule mounted on the ROV with camera and light facing downward (left)

COMPUTER VISION OVERVIEW

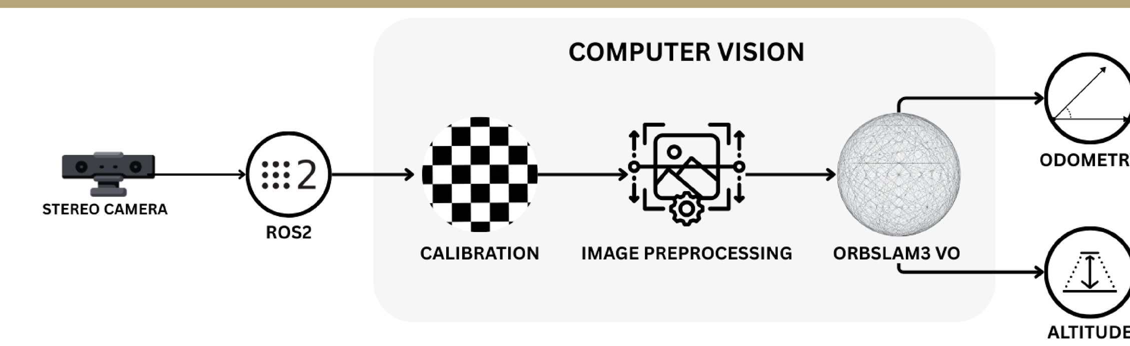

Figure 3. Computer vision pipeline, takes in stereo camera feed and outputs odometry

CALIBRATION & PREPROCESSING

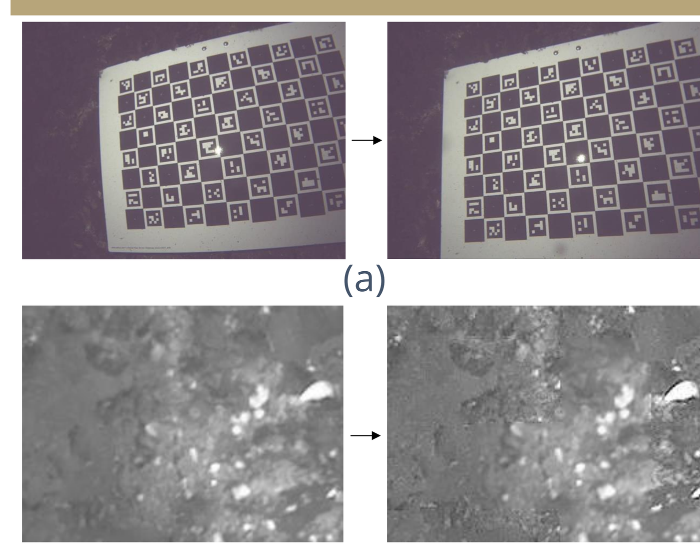

Figure 4. (a) Rectified and (b) pre-processed images used by ORB-SLAM

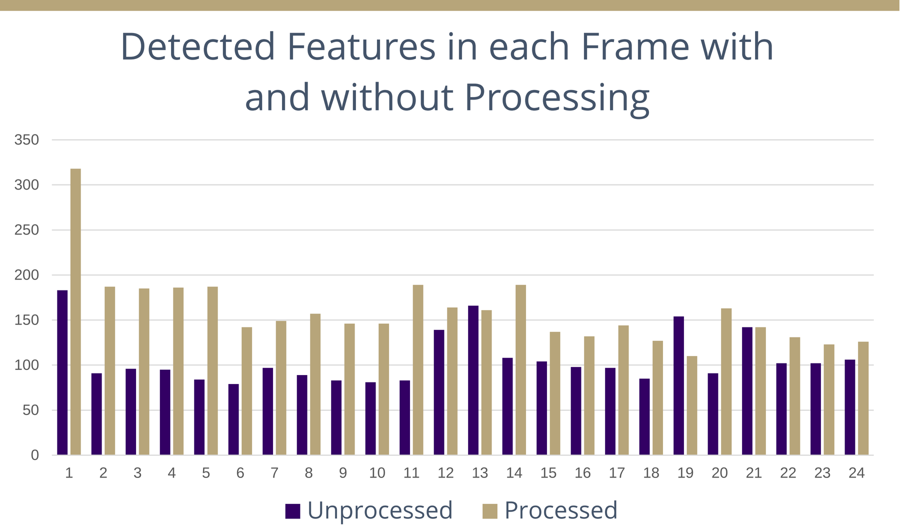

Figure 5. Comparison of detected features after processing the images with information entropy and adaptive sharpening

ORB-SLAM3 & ODOMETRY

ORB-SLAM3

- A VSLAM program that receives stereo images and output relative pose between cameras and the world
- Fine-tuned to function better in near-seabed underwater environment

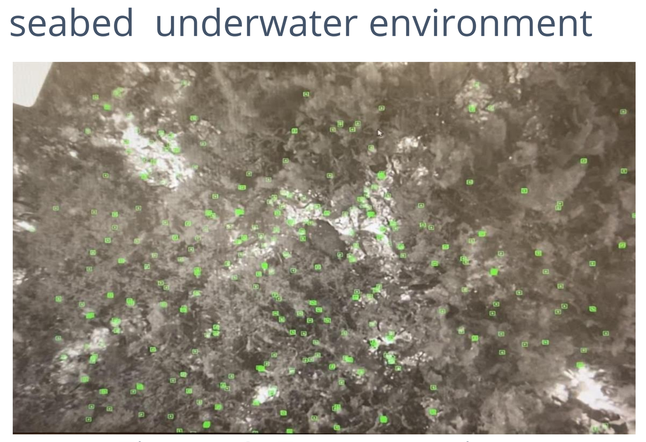

Figure 6. Feature points

ODOMETRY WRAPPER

- Establish ROS data communication method for ORB-SLAM3's in/output
- Fetch the pose of map point from ORB-SLAM3 and publish position, orientation, linear and angular twist with covariance
- Estimate the altitude using 3D point cloud

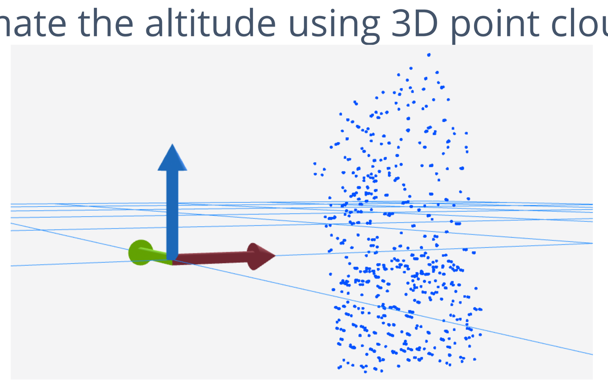

Figure 7. 3D point cloud

MAVLINK

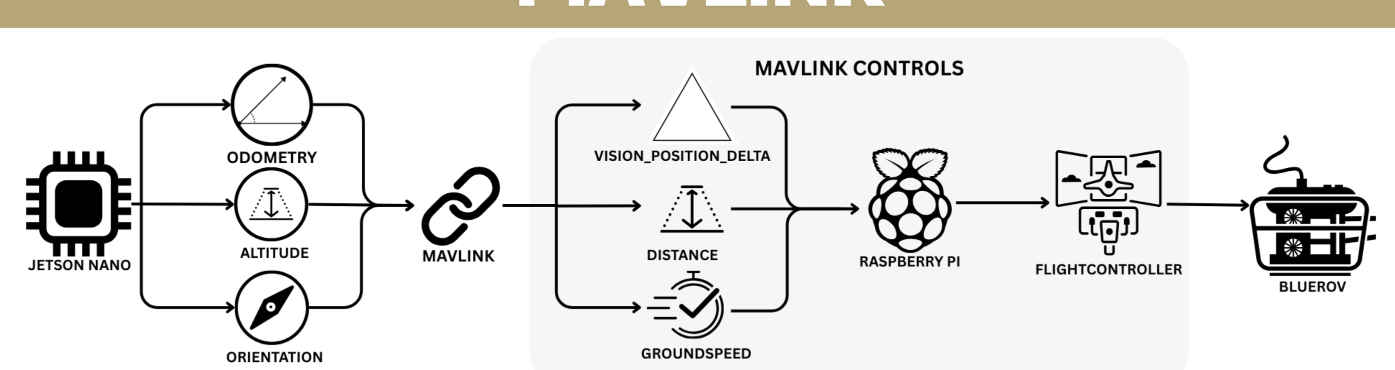

Figure 8. MAVLink pipeline with our Nano and the internals of the ROV

RESULTS

PLOTTED TRAJECTORY AND ODOMETRY FROM TESTS

- Generated point cloud of features, while plotting the traversed trajectory
- Output pose, twist, and depth to ROS topics
- MAVLink displays values from topics into QGroundControl

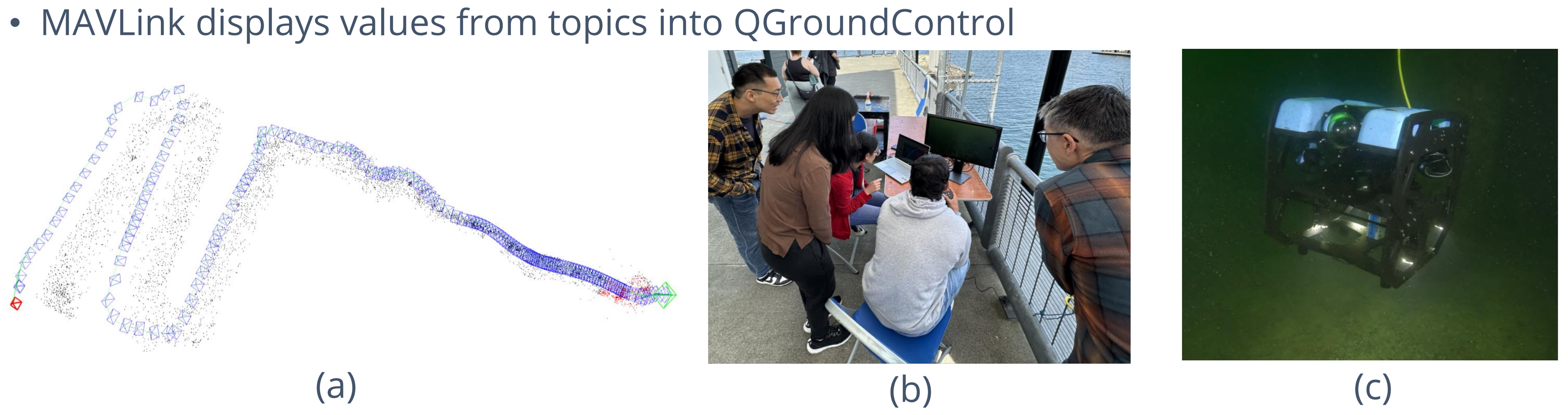

Figure 9. Test running the entire system in real-time at the pier (b and c) and the trajectory of ROV estimated by ORB-SLAM (a)

VISION POSITION DELTA

Our Visual Odometry implementation showed promising results for sending velocity data, but showed higher noise, lower reliability, and increased outliers compared to the DVL.

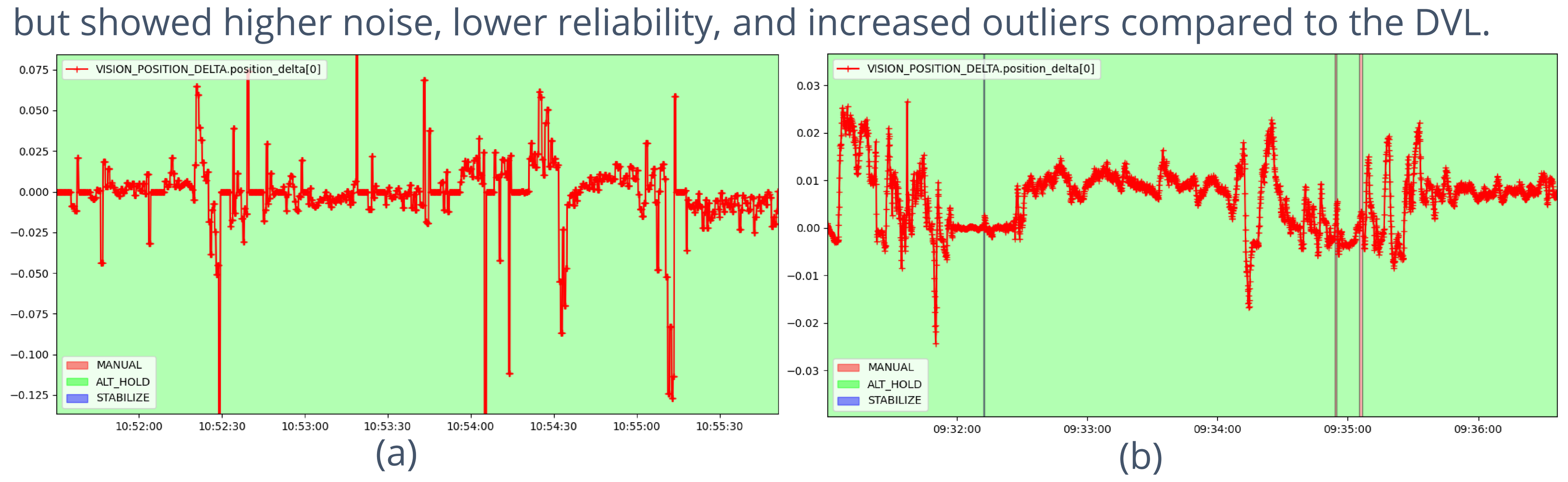

Figure 10. (a) Visual Odometry Implementation Test Results and (b) DVL Results

DISTANCE SENSOR

Our implementation showed consistent results in sending altitude values but dipped to zero centimeters at times. This was due to ORB-SLAM3 Odometry not detecting features.

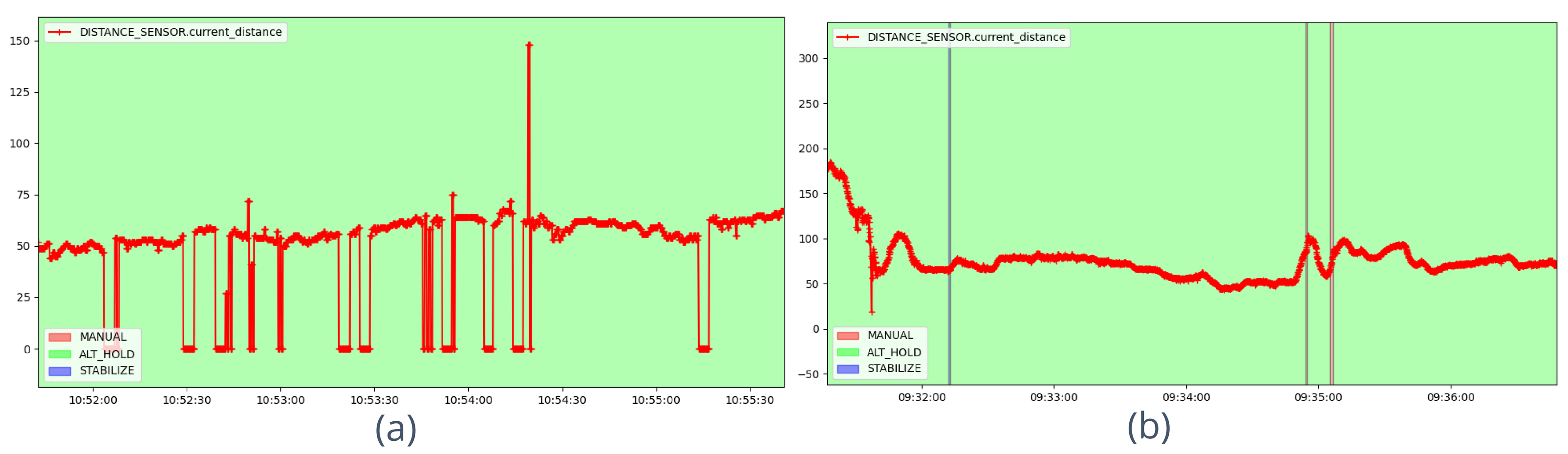

Figure 11. (a) Visual Odometry Implementation Test Results and (b) DVL Results

ORBSLAM achieved a relative error of 4.85% relative to the speed ground truth, compared to 4.42% for the DVL. This indicates that the implementation is moving in the right direction and could closely approach DVL-level performance with further parameter optimization.

CONCLUSION & FUTURE WORK

This project demonstrates a live underwater visual odometry and communication system by integrating visual SLAM and MAVLink messaging on a BlueROV2 platform.

The next steps for this project will be to:

- Improve accuracy of the system by ensuring parameters are fully tuned
- Implement a functioning ROS2 – ORBSLAM3 VIO system with a fully calibrated system
- Implement with CUDA to increase the performance

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISERS: AARON MARBURG, ZACHARY RANDELL, CLYDE MCQUEEN

SPONSORS: SEATTLE AQUARIUM, UW APPLIED PHYSICS LABORATORY