

# MUN RoboSub Technical Report:

## Memorial University of Newfoundland



The MUN RoboSub project team was started in September 2021 and has grown to a sizeable number of students working to develop the first version of the AUV which will represent the University at the annual RoboSub competition. For simplicity and clarity of roles, the team subdivided into three technical subsystems; mechanical, electrical and software. The software subsystem was further divided into two: computer vision and navigation teams. While both are software groups, the navigation team is responsible for integrating sensory input and deciding the appropriate motor output for the developed mechanical system. On the other hand, the computer vision team implements image processing and recognition, and machine learning. Each of the subsystem's work is further elaborated as chapters in the report, starting with mechanical, electrical and then the two software aspects. This was the first time that the group competed in the competition and the team is excited to see what it can be able to achieve.

### 1. Mechanical

#### a. Housing/Enclosure

The team decided to focus on functionality and made a simplistic housing design which would house all the components that would be needed in terms of electronics and cameras as well as sensors that would be used inside the housing. For this, the team went with an off the shelf Blue Robotics cylinder which would house most of the components including the camera, the batteries and other circuitry. Consultation was made with

the electrical team to determine the sizes of the components before the decision was made to purchase an 8-inch diameter acrylic cylinder which was 2 feet long. The following is an image of the cylinder.



### b. Electronics Trays

To house the components inside the housing described above, a system of trays was designed which would simplify the process of taking out components from the AUV. These were designed based on the geometry of the components that would be used inside the AUV and were 3D printed by the team. The opportunity to 3D print these trays made it possible to customize the design to one that would allow the components to sit comfortably and be located at optimum locations, such as the batteries at points where they can be easily be taken out for recharging or the computer for loading new info.

### c. Frame

Initially the team started with an enclosed design that would mimic deep-water AUVs which would be either cylindrical or spherical in nature. However, given the small depths the AUV would experience of a maximum of 4 feet according to competition guidelines, the team decided to focus on making a reconfigurable frame design. This design would make it easy for future groups to improve upon the AUV. Thus the team made a frame design from aluminum members that are joined together using nuts and bolts. This gives the team the unique ability to modify its frame shape when the need arises, which is useful at this initial stage of the team's development since many iterations of the design are anticipated.

### d. Thruster Shrouds

As per competition guidelines and for human safety and protection of the thruster propellers, thruster shrouds were to be manufactured which were used to protect the T200 thrusters

### e. Camera Mounts

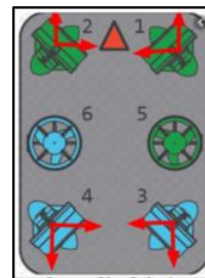
To position the cameras that would be used by the AUV, camera mounts were made which would help position the camera in the correct position. The design was a simple one for both the forward facing camera and the downward facing camera.

### f. Cylinder Brackets

To attach the enclosure cylinder to the frame, brackets were used to hold it in place. These brackets were customized to the shape of the cylinder and were used together with nuts and bolts to hold the cylinder in place. The advantage of this design was that it allowed for the future configuration of the frame and cylinder positions.

### g. Thruster Configuration

To achieve the required degrees of freedom, the team did some research. The aim of this research was to find an option that would make for a simple initial configuration which was compatible with the software that would be used for the AUV. In the end the team went for a configuration of 6 thrusters which would give us 5 DOF.



### h. Buoyancy

The team made use of small weights (1lbs, 2lbs etc.) to actively manipulate the weight of the AUV. This helped to increase the weight of the AUV to reduce its positive buoyancy.

## 2. Electrical

### a. Wiring Diagram

The first iteration of the wiring diagram was created in SolidWorks Electrical to illustrate how different components on the AUV connect and communicate with each other. The wiring diagram will be maintained and updated as component modules are changed, and improvements are made to the AUV's systems. The main electrical systems are divided into propulsion, powers distribution (power supply) and onboard electronics.

### b. Propulsion

The AUV uses six Blue Robotics T200 thrusters which are specially designed for efficient underwater applications. Each thruster can generate forward and reverse thrust and are accompanied by electronic speed controllers that accept pulse width modulation (PWM) signals from the Pixhawk flight controller.

The value of the PWM signal controls the magnitude and direction of thrust force generated by the thruster.

#### *c. Power Distribution*

The main objective of the power supply module will be to distribute the appropriate amount of power (voltage & current) to each of the other major electrical systems (propulsion and electronics). The main source of power is via six Li-Po (Lithium polymer) batteries with two connected to a power distribution board (PDB) with voltage conversion capabilities. This PDB powers two thrusters as well as some of the electronics components on the AUV. The other four batteries are dedicated for thruster power. Modular connectors will be implemented to interface the batteries with the PDB, and this interface will serve as the main system switch.

The purpose of the distribution board is to serve as a junction where power can be routed from the battery to the electrical load of the AUV. Integrating a board with built-in conversion capabilities is beneficial mainly because it saves spaces and removes the need for additional conversion stages in the system. Batteries are also implemented with 30 Amp fuses to limit the current input to the PDB such that any unexpected fluctuations in current draw have minimal effect on the load.

#### *d. Electronics*

The electronics module includes a range of sensors and computing systems that help the AUV achieve autonomy. These elements will generally be running on lower voltages and currents compared to the thruster motors.

### **3. Software**

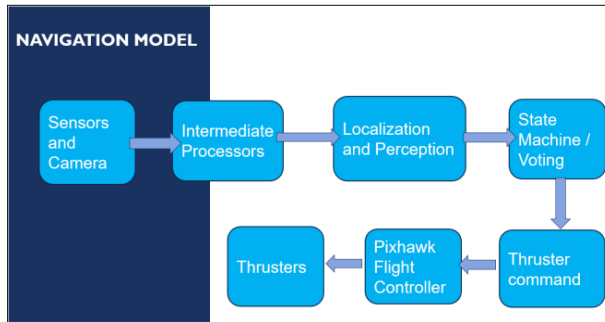
As mentioned earlier, the software group was divided into two teams, the navigation team and the computer vision team.

#### **3.1. Navigation Team**

##### *a. Path Planning*

The Robot Operating System (ROS) software was used to develop the navigation algorithm. The publisher-subscriber model from ROS was used for communicating with hardware and software components of the AUV. The navigation processes will involve the following steps:

- *Extracting Information:* from sensors and visual data from the camera
- *Intermediate Processors:* to send this collected data to the Jetson computer on board
- *Localization and Perception:* where the collected data is processed in the computer to obtain position estimates
- *Voting Process:* reads the position estimates from the localization algorithm, and acts as a failsafe if one of them fails. The voting algorithm chooses sensor information it deems most appropriate at the time and sends it to the thruster command process.
- *Thruster Command Process* aims to move the AUV to a desired waypoint. Using the localization algorithm, the process first tells the AUV to orient itself towards the target using the compass
- *Pixhawk Flight Controller:* receives simple translation commands and sends them to the thrusters which drives the AUV to the desired location.



### *b. Development*

The main tasks that the team planned on finishing consisted of developing a control algorithm (namely go-to-goal) to move the AUV from its current location to a desired target location, and to use the acceleration (and angular acceleration measurements) in 6 DOF to estimate the current position of the AUV.

Attitude estimator which is a generic platform-independent C++ library to implement an IMU sensor fusion algorithm will be used for estimating the 3D attitude (orientation) estimate in the form of quaternion.

Go-to-goal algorithm that moves the AUV from its current position to a target position uses Feedback control to adjust the AUV's translational and angular velocity dynamically, using only the desired location and distance tolerance as inputs. Here, distance tolerance implies to the difference in the current position and the final target position. It is set to a very small constant value of 0.01 for these simulations.

## **3.2. Computer Vision Team**

### *a. Background Research*

Based on the competition guidelines for the tasks, the AUV must identify a gate, pass through one of its sides, and identify characters such as bootlegger / G-man.

Therefore, the background research on determining the best approach for object detection in the AUV's computer vision system focused on published past competitor's reports

and the conclusion deduced was to use a neural network framework for classification in conjunction with a color thresholding method for more robust detection. Additionally, preliminary research also suggested that implementing multiple detection techniques is more beneficial than relying on one method.

OpenCV is a very popular tool for image processing and performing computer vision tasks. It also supports python. Hence it was chosen for image/frame processing, visualization of results and adding more detail (draw lines, boundaries etc.) to an image or frame as necessary. Processing enhances detection, particularly through thresholding and segmentation of the object of interest from the rest of the background where possible. However, this depends on the object of interest being detected in an instance.

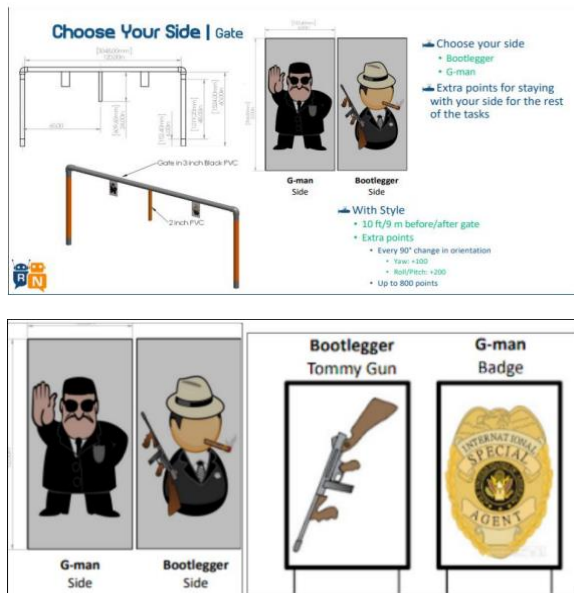
Due to the time constraint, it did not seem feasible to make a neural network framework from scratch and You Only Look Once (YOLO) repeatedly showed up in past competitors' reports. Therefore, YOLO was further investigated. YOLO models are infamous for being highly performant and fast yet incredibly small – making them ideal candidates for real-time conditions and on-device deployment environments. YOLOv5 was chosen as this is the first native release of models in the YOLO family to be written in PyTorch first rather than PJ Reddie's Darknet.

There are various pre-trained YOLOv5 models to choose from, out of which the YOLOv5s seemed most suitable to our needs. It's the smallest and fastest model.

The basic color thresholding technique chosen was the Hue Saturation Value (HSV) color space because of its relative illumination invariance property when compared to the RGB space. The HSV color space allows the segmentation of a particular color by giving the ability to specify color ranges independent of the brightness or luminance value.

### *b. Design Approach to Algorithms*

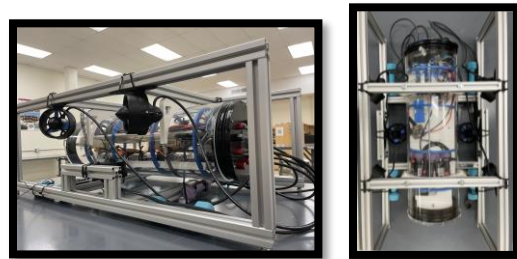
The dataset consists of six classes: 'smallGate', 'Gate', 'gMan', 'bootlegger', 'tommyGun' and 'badge'. Below is a picture of the classes:



To train the custom YOLO model in Pytorch, it was necessary to collect, organize and assemble a dataset of representative images of our classes with bounding box annotations around the classes that are required to be detected.

There were two methods of preparing the dataset manually on an annotate tool or using Roboflow.

Creating our own dataset and labels proved to be challenging as applying any pre-processing and augmentation techniques would have too many moving parameters. Hence the team decided to use Roboflow, an integrated software available for free to students. Since the competition themes change year to year, it would also be beneficial to easily introduce any junior students to the computer vision system, whilst making them feel engaged and impactful. The hope is this will motivate them to dig deeper into neural networks and machine learning.



## Appendix A: List of Components

Component	Vendor	Model/Type	Custom/Donated	Cost (CAD)
Frame		Aluminum	Donated	
Waterproof Housing	Blue Robotics	8-in Enclosure x1		240
Waterproof Penetrators	Blue Robotics	Wetlink Penetrators x 10		130
Thrusters	Blue Robotics	T200 x6		1620
Motor Control		Pixhawk Controller Advanced 32 bit ARM Cortex® M4 Processor running NuttX RTOS		150
ESCs	Blue Robotics	Bidirectional Basic ESC x 6		270
Batteries	SIGP	Rechargeable LiPo batteries 4S 14.8V 7200mAh battery x 6		324
Battery Charger	Radiolink	CB86-PLUS		100
CPU	NVIDIA	Jetson Nano Computer		210
IMU		Onboard Pixhawk		
Camera	Stereolabs	ZED stereo camera 2(380mA / 5V USB Powered)		500
Housing Mounts			Custom 3D printed	
Buoyancy weights	Canadian Tire	1lb and 2lb weights		70