Ecolymer River Cleanup Robot

Systems Integration of an Unmanned Surface Vehicle for Autonomous Removal of Plastic Waste in Aquatic Environments

ECE 4872: Final Semester ECE Senior Design
9/21/2021

ELECTRICAL [+] COMPUTER

E N G I N E E R I N G

Project Background

- **United Nations Sustainable Development Goal 14.1**
  - Challenges the world to ”prevent and significantly reduce marine pollution of all kinds, in particular from land-based activities” by 2025 [1].
- Oceans produce over half the world’s oxygen as well as having an economic impact of **$282 billion** in the U.S. alone [2].
- River systems have a significant impact on this flow of waste as it is estimated that just **1000 rivers contribute 80%** of global annual emissions up to the magnitude of 2.7 million metric tons per year [3].

Mekong River [4]

[1] https://sdgs.un.org/goals/goal14
[2] https://oceanservice.noaa.gov/facts/why-care-about-ocean.html
[3] https://theoceancleanup.com/sources
[4] https://www.independent.com/2019/06/02/new-program-targets-down-river-plastic-waste/

Our Team

Tan Tonge
Project Manager

EE - Specializing in Smart Power Grid Control / Renewable Energy Integration

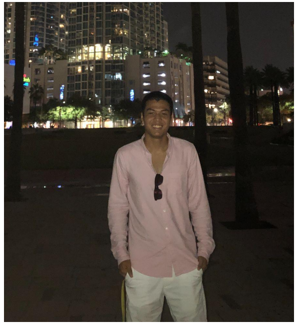
Jin Bae
Navigation Software Lead

CompE - Specializing in Embedded Systems/Robotics

Luis Pimentel
Perception Software Lead

CompE - Specializing in Robot Control Systems/Machine Learning/Perception

Alexander Chanthaphaeng
Hardware Lead



EE - Specializing in Electronics / Hardware Design

Sponsor Needs

- Ecolymer desires a **fully-automated** river clean-up system
  - Intended to be deployed in waterways to:
    - **Identify** plastic waste
    - **Collect** the waste
    - **Dispose** of the waste
  - Without interfering with river traffic or aquatic life

- Developing a prototype proof-of-concept of their desired solution
  - Focusing on the **limiting technologies** to accomplishing these operations

Design Objectives

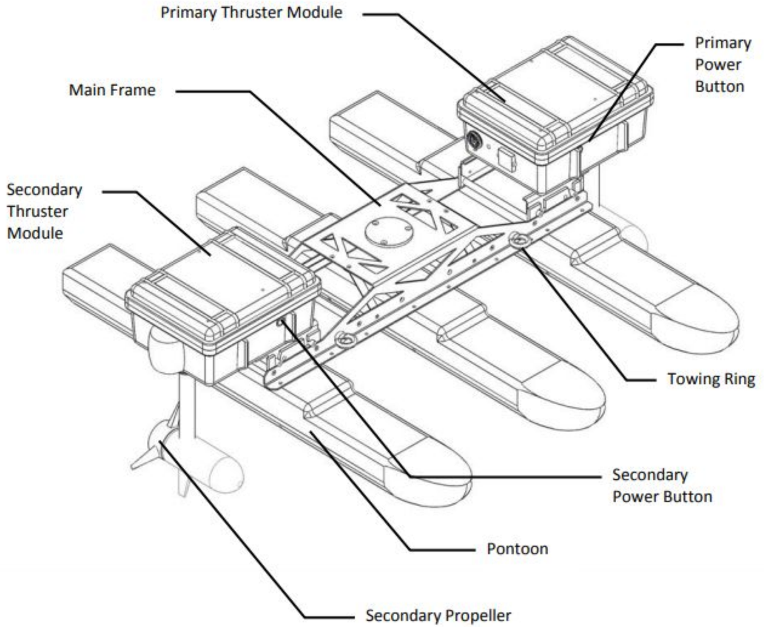
- Continuing the development from 2 prior semesters of Senior Design
  - Fall 2020 - Trash Detection through YOLO
  - Spring 2021 - Collision Avoidance with Xbox Kinect
  - **Fall 2021 - Integration onto USV platform to achieve autonomous navigation**

- Leveraging existing Unmanned Surface Vehicle (**USV**) platform to avoid **hardware development** time
- Integrating commercial and **readily-available** solutions
  - Reduce costs and improve feasibility of final product

Explanation of Design

Base Robot Platform

- **Kingfisher M100** has been identified as the proof-of-concept USV platform



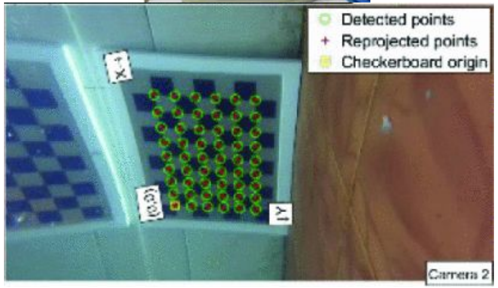
Kingfisher M100 and its components [1]

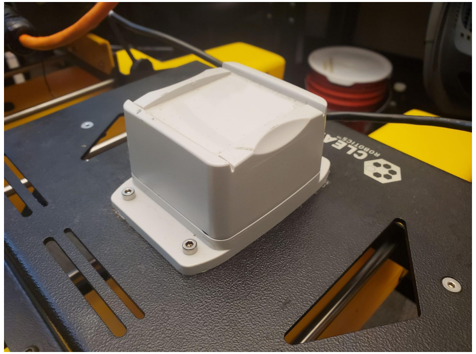
[1] https://oceanai.mit.edu/svn/moos-ivp-kfish/trunk/docs/kfish-m100-manual.pdf

Sensors

ZED 2i [1]

- **GPS Module:** vehicle’s position in body of water
- Inertial Measurement Unit **(IMU):** vehicle’s orientation
- **ZED 2i:** stereo vision camera for trash detection and depth sensing
  - To be placed in a forward-facing waterproof enclosure [1]



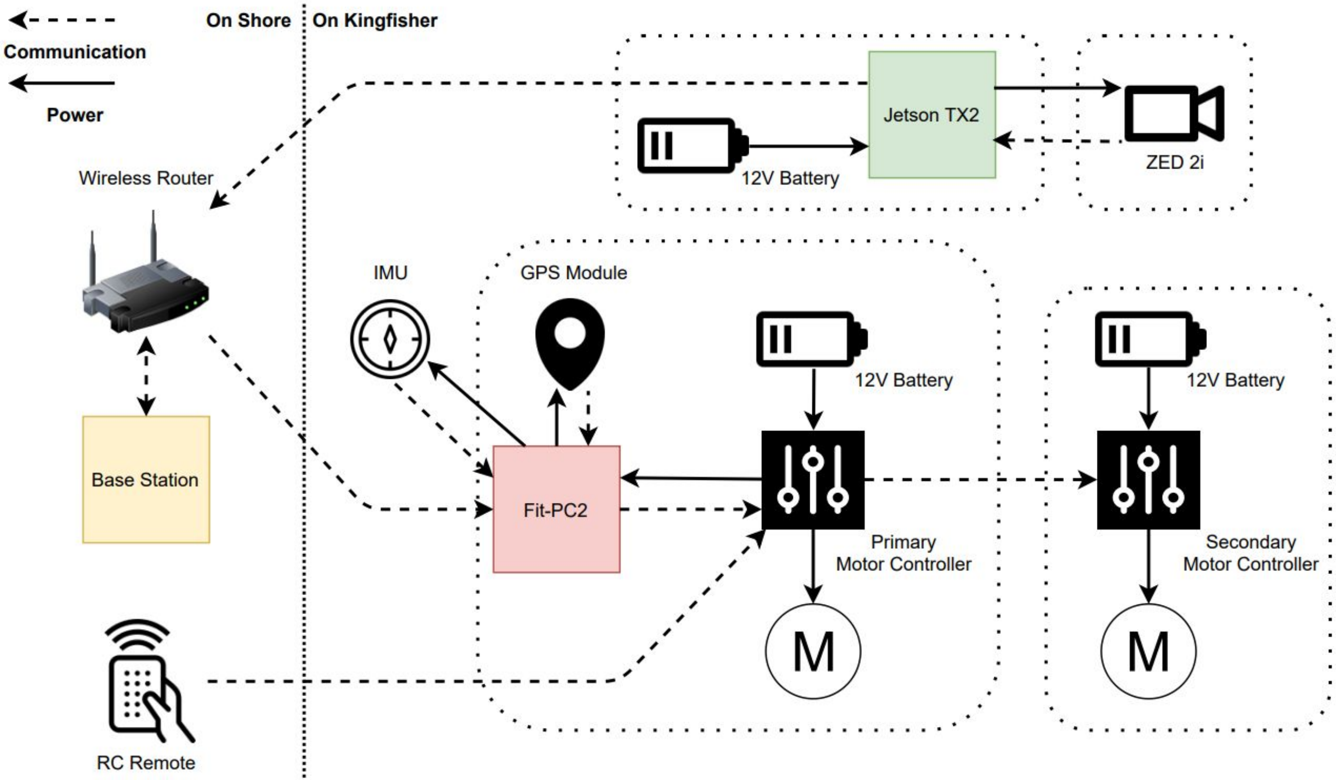


IMU

GPS

[1] https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8867236

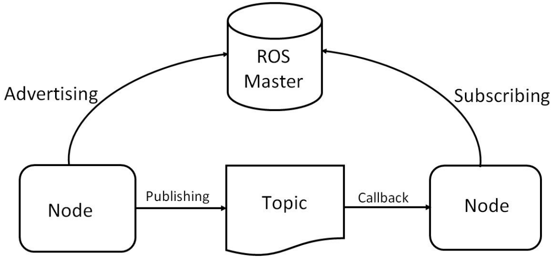
Hardware Architecture

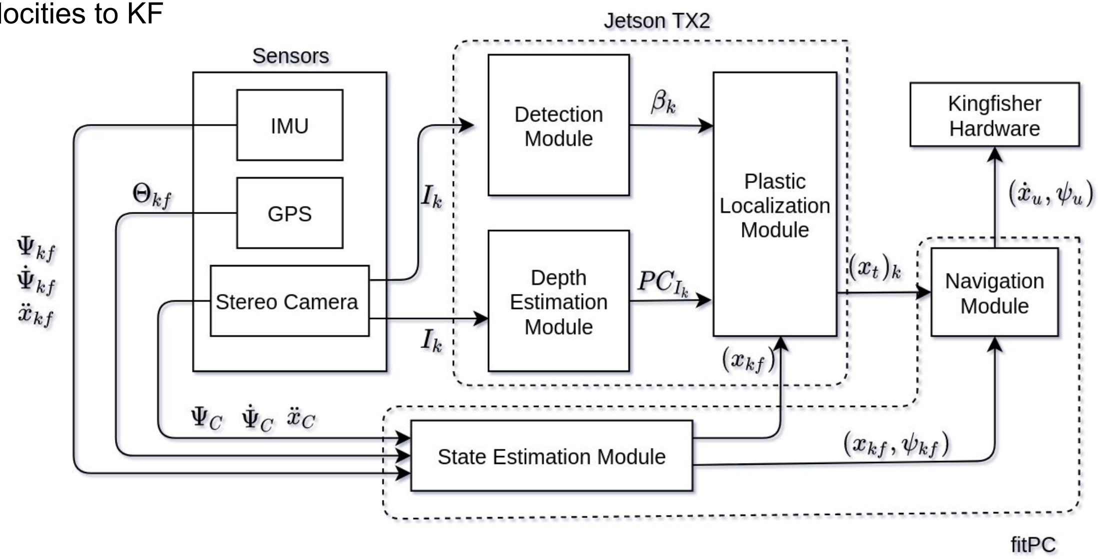


Software Architecture

- Our software architecture consists of 5 main modules:
  - **Detection Module**: detects plastic in an image frame
  - **Depth Estimation Module**: outputs detected 3D point-cloud data
  - **Plastic Localization Module**: estimate the positions of the trash
  - **State Estimation Module**: outputs robot state estimates
  - **Navigation Module**: outputs commanded velocities to KF

- Utilizing the Robot Operating System (ROS) [1] for communication between modules



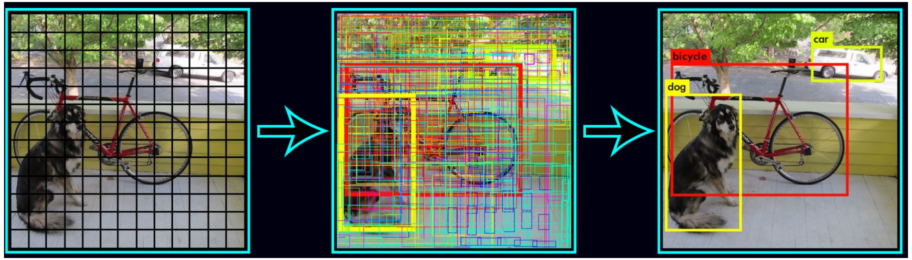


STEREOLABS ROS

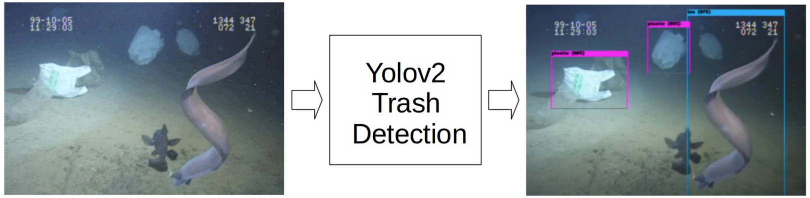
[1] https://www.ros.org
[2] https://www.designnews.com/gadget-freak/ros-101-intro-robot-operating-system

Perception: detection of underwater plastic waste

- Underwater plastic water will be detected using a real-time performing object detection system: Yolov2 [1].
  - Runs images through a neural network with the ability to **classify and localize** an object
  - Outputs the **bounding boxes** (relative pixel locations) around the detected object



- The pre-trained weights on the trash_ICRA19 [2] dataset will be used
  - Trained to detect three object classes: **plastic**, marine biology, and ROV (remotely operated vehicle)

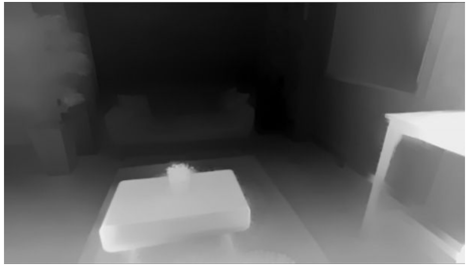


Input image frame   Output image frame

[1] https://pjreddie.com/darknet/yolov2/
[2] https://conservancy.umn.edu/handle/11299/214366

Perception: Depth Estimation

- ZED 2i stereo camera captures images of the environment used by the Detection Module and Depth Estimation Module
  - StereoLabs provides APIs to access its internal depth estimation modules providing:



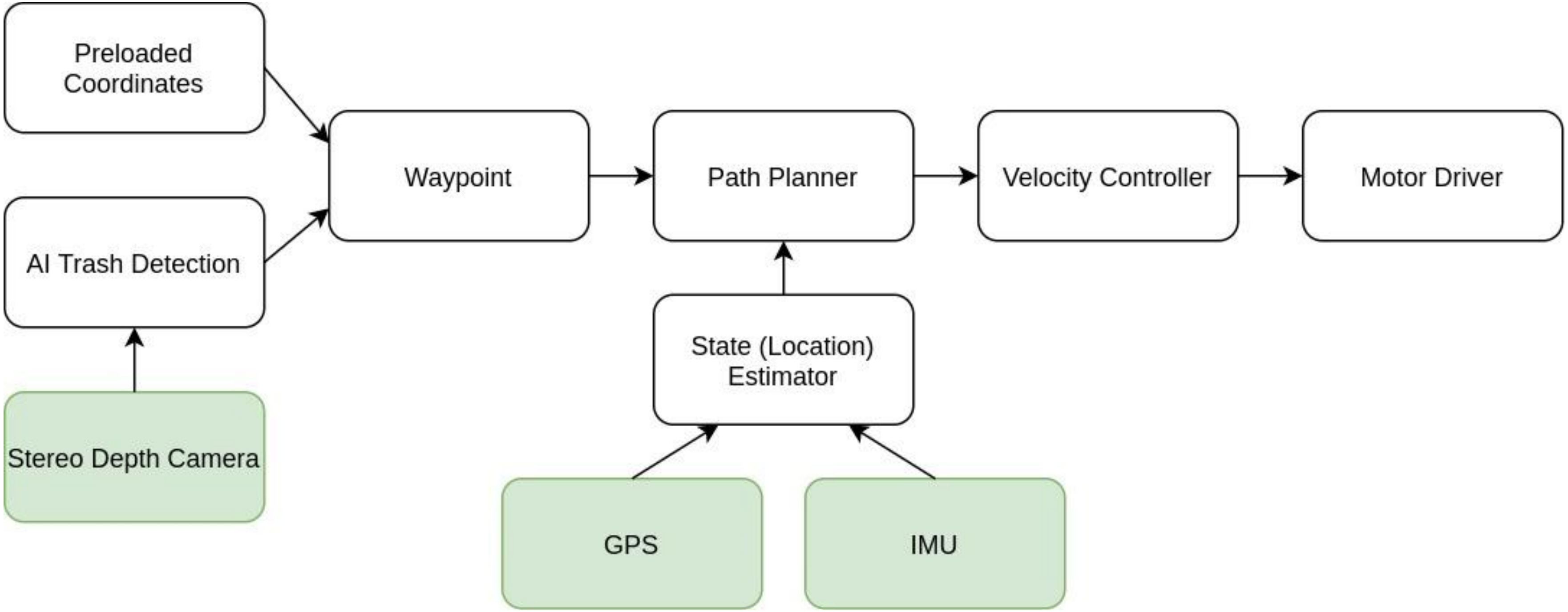
Depth Estimation Data
scalar distances to objects in image frame

Live Point-cloud Data
3D points relative to the location of the camera

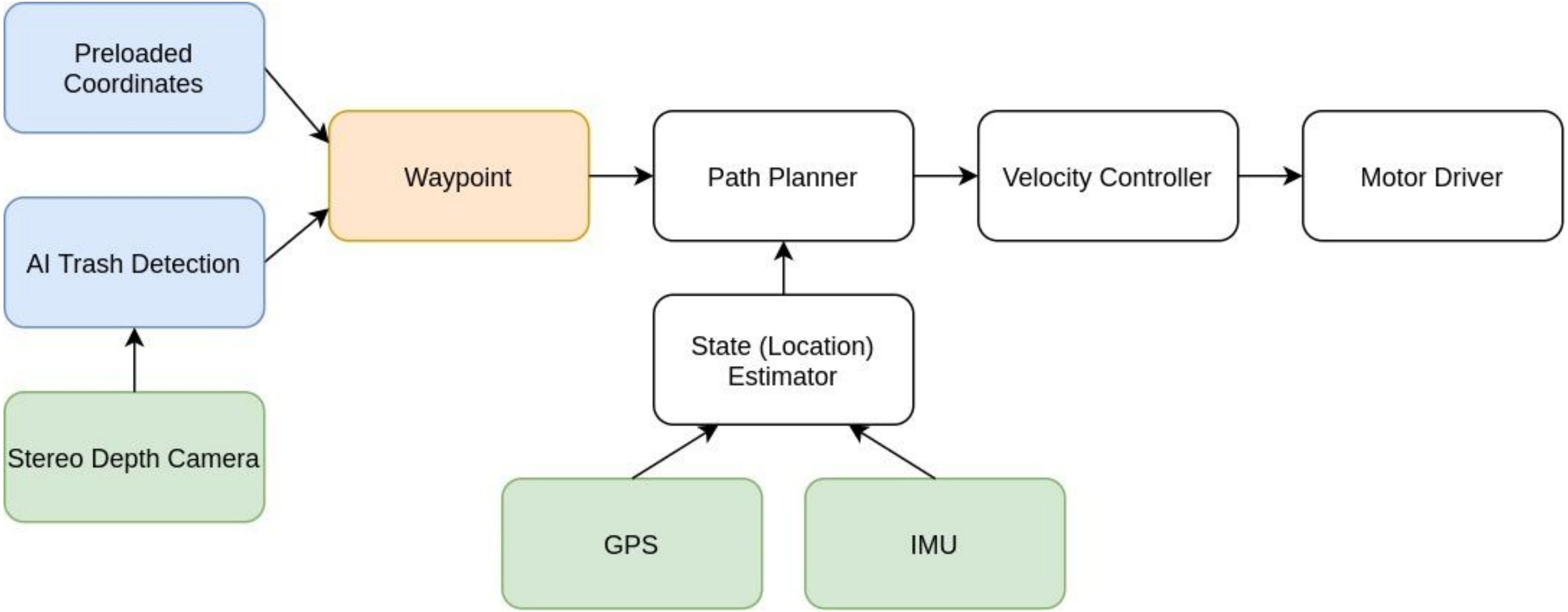
- We can then cluster the 3D points perceived within the K bounding boxes detected
- We can formally define this as a K-means clustering problem due to possible overlap in bounding boxes that:
  - assigns 3D points to K clusters
  - estimates the mean 3D location

$$\underset{\mathcal{C}}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{{}_W\mathbf{x}_s^b \in \mathcal{C}_k} \| {}_W\mathbf{x}_s^b - u_k \|_2^2 \qquad u_k = \frac{1}{|\mathcal{C}_k|} \left(\sum_{{}_W\mathbf{x}_s^b \in \mathcal{C}_k} {}_W\mathbf{x}_s^b \right)$$

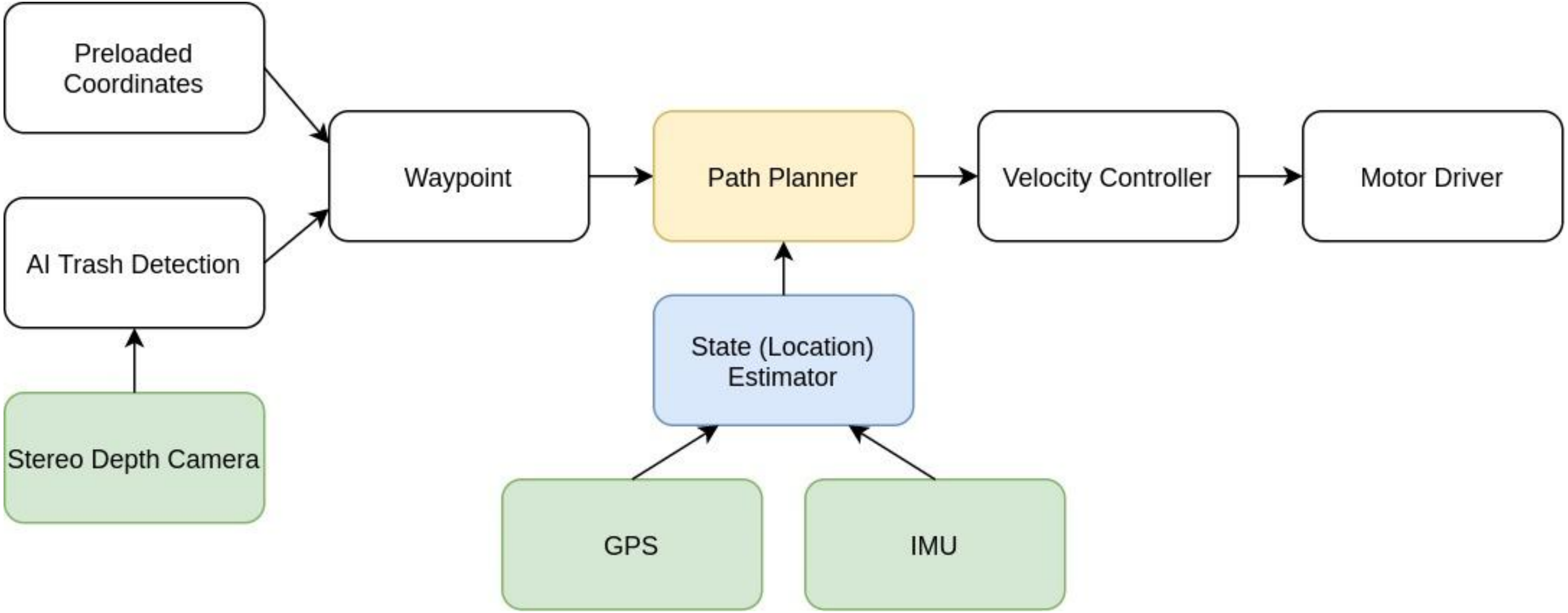
Navigation Pipeline



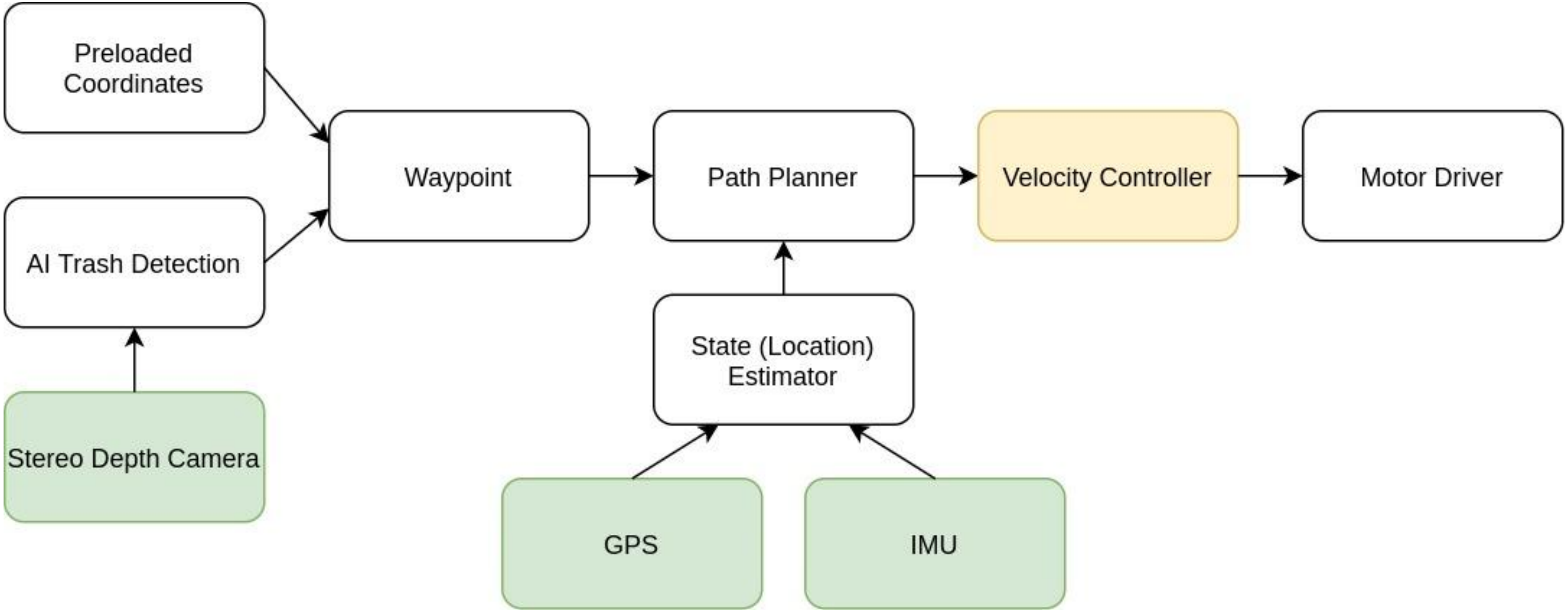
Navigation Pipeline



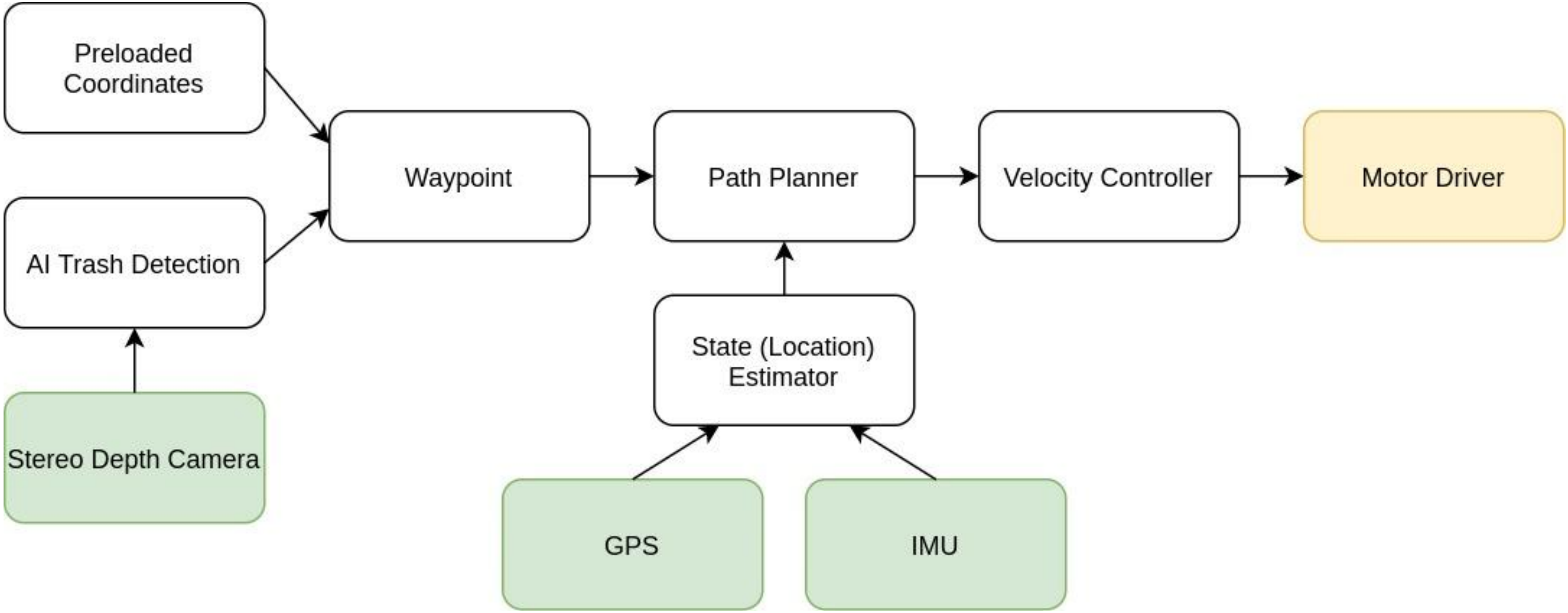
Navigation Pipeline



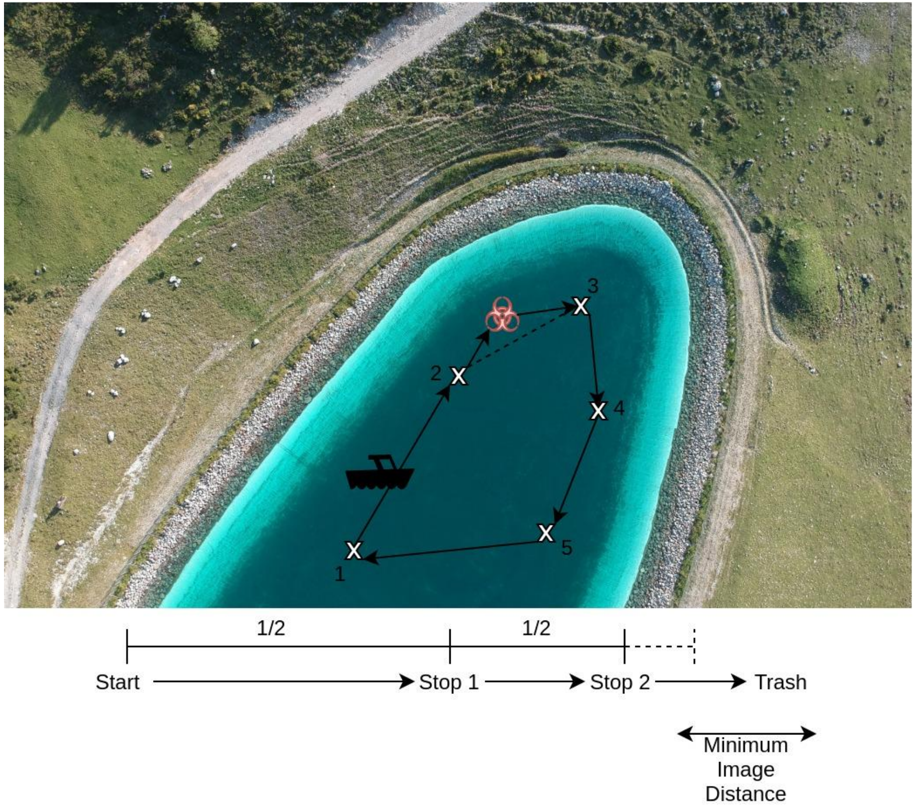
Navigation Pipeline



Navigation Pipeline



Example Navigation Sequence



waypoints.csv

1. 33.74905° N, 84.38801° W
2. 33.74907° N, 84.38810° W
3. 33.74918° N, 84.38813° W
4. 33.74924° N, 84.38808° W
5. 33.74917° N, 84.38802° W

Project Specifications

Qualitative Goals

- Ability to **detect trash** objects using camera system and pre-trained trash detection model
- Ability to **localize trash** using bounding box and its corresponding depth values
- Ability to control the robot's position and heading through ROS
- Ability to **navigate robot** from a starting point to a trash location
- Ability to continuously estimate trash location as trash location drifts
- Ability to operate robot for given operational time

Quantitative Specifications

- Ability to detect all trash within 3 meters with confidence p >= 0.8
- Ability to detect trash locations of:
    - 100% <= 3 meters
    - 60% 3 meters < x <= 10 meters
    - 20% 10 meters < x <= 20 meters
- Ability to navigate from starting point to trash locations:
    - 100% <= 3 meters
    - 60% 3 meters < x <= 10 meters
    - 20% 10 meters < x <= 20 meters
- Ability to continuously estimate trash location with drift in location of 1 meter
- Ability to operate vehicle for 4 hours

Detection Integration Testing

- The Idea:
    - Integrate the ZED 2i camera ensuring successful image streaming capabilities on the Jetson TX2 and perform real-time object detection
- Procedure:
    - Install libraries and dependencies to stream data from the camera to the Jetson.
    - Mount ZED 2i in a transparent waterproof enclosure, calibrate camera, and validate trash detection underwater.
    - Mount camera with enclosure on the Kingfisher and place in pool with sample trash objects and non-trash objects at different known locations and different ranges, validating accuracy.

Depth Estimation Testing

- The Idea:
  - Using the ZED2i camera’s depth capabilities, estimate the 3D locations of trash.
- The Procedure:
  - Develop depth estimation pipeline with ZED 2i camera and validate in lab.
  - Install camera in a transparent waterproof enclosure and validate depth estimation in a water tank.
  - Mount camera with enclosure on the Kingfisher and place in pool with with sample trash objects at different known locations and different ranges, validating accuracy of depth estimation.

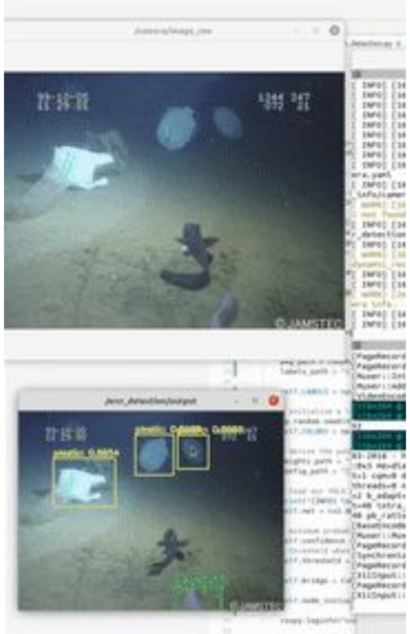
Collision Avoidance and Navigation Testing

- The Idea:
  - Use the collision avoidance and navigation from Spring 2021 team.
  - Test navigation and collision avoidance capabilities in a water environment before full systems test.
  - Transform and navigate coordinates in the local frame, using global GPS frame.
- Procedure:
  - Have obstacles in a pool or lake environment (buoys, tree branches, makeshift boats).
  - Run collision avoidance and navigation on Kingfisher from Spring 2021 team.
  - Navigate Kingfisher in the water.

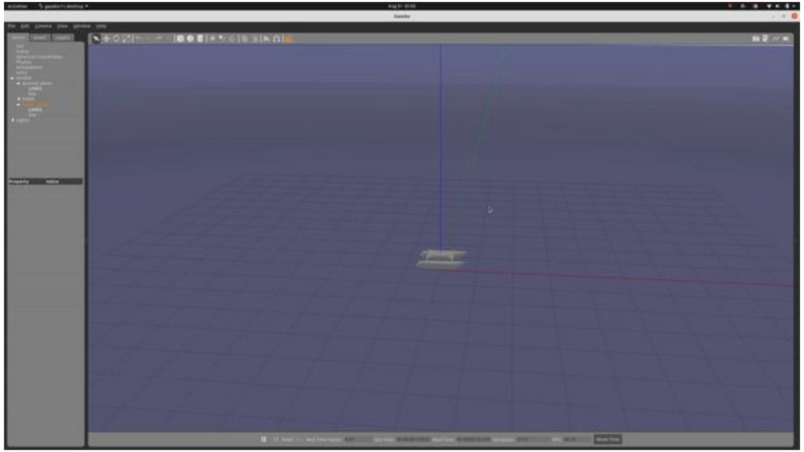
Current Project Status

Current Status

- Have new fitPC to replace original with broken port
- Functioning ROS node doing object detection on real-time video stream using Yolov2
    - Image frame detection comes in at ~2Hz, likely sufficient for our application
    - Detection at ~9Hz using tiny_yolo. Less accurate, more false positive detections



- Gazebo simulation of a Autonomous Surface Vehicle (ASV) using latest ROS/Gazebo libraries
    - Implemented a thruster model, dynamics model, and sensor models (IMU and GPS)
    - Allows us to test code that will can drive the robot through ROS
    - Adapted from Kingfisher_gazebo [1] and usv_gazebo_plugin [2].



[1] https://github.com/bsb808/kingfisher_gazebo
[2] https://github.com/bsb808/usv_gazebo_plugins

Project Schedule

WBS	TASK	LEAD	START	END	DAYS	WORK DAYS
1	**Existing System Integration**			-		-
1.1	Existing Hardware Evaluation	Alex	Wed 8/25/21	Thu 8/26/21	2	2
1.2	MOOS Evaluation	Jin	Mon 8/23/21	Sun 8/29/21	7	5
1.3	YOLO Evaluation	Luis	Mon 8/23/21	Sun 8/29/21	7	5
1.4	Tele-Op Test Driving	Tan	Wed 9/08/21	Wed 9/08/21	1	1
1.5	Mount and power Jetson	Tan	Mon 9/06/21	Sun 9/12/21	7	5
1.6	Setup Jetson TX2 software	Luis	Mon 9/06/21	Sun 9/12/21	7	5
1.7	Setup new fitPC	Jin	Mon 9/06/21	Wed 9/08/21	3	3
2	**New System Implementation**			-		-
2.1	Implement Trash Detection Software	Luis	Mon 8/23/21	Sun 9/05/21	14	10
2.2	Install camera enclosure and mount	Tan	Mon 9/13/21	Sun 9/19/21	7	5
2.3	Setup camera and software libraries	Luis	Mon 9/13/21	Sun 9/19/21	7	5
2.4	Navigation Software for Kingfisher	Jin	Mon 9/06/21	Sun 10/03/21	28	20
2.5	Implement Trash Detection and Localization using camera	Luis	Mon 9/13/21	Sun 10/03/21	21	15
2.6	Develop Software between Detection and Navigation	Jin	Mon 9/27/21	Sun 10/10/21	14	10
2.7	Setup networking communications	Alex	Mon 9/27/21	Sun 10/10/21	14	10
2.8	Full Sytems Integration	Luis/Jin	Mon 10/11/21	Sun 10/31/21	21	15
2.9	Bug Fixing/Troubleshooting	Luis /Jin	Mon 11/08/21	Sun 11/14/21	7	5
3	**Testing**			-		-
3.1	Develop Testing Benchmarks	Alex	Mon 10/18/21	Sun 10/24/21	7	5
3.2	Prepare Test Trash/Objects	Alex	Mon 10/25/21	Sun 10/31/21	7	5
3.3	Testing Round 1	Tan	Mon 11/01/21	Sun 11/07/21	7	5
3.4	Testing Round 2	Tan	Mon 11/15/21	Sun 11/21/21	7	5
4	**Expo/Deliverable Preparation**			-		-
4.1	Revised Proposal	Tan	Mon 9/06/21	Sat 9/25/21	20	15
4.2	Proposal Presentation	Tan	Mon 8/30/21	Sun 9/05/21	7	5
4.3	Poster	Tan	Mon 11/15/21	Sun 11/21/21	7	5
4.4	Website	Jin	Mon 11/22/21	Sun 11/28/21	7	5
4.5	Project Report	Alex	Mon 11/01/21	Tue 11/30/21	30	22
4.6	Practice Presentation	Tan	Mon 11/29/21	Sun 12/05/21	7	5
4.7	Capstone Expo	Tan	Tue 12/07/21	Tue 12/07/21	1	1

Questions?