EE 175-022
Team 24

Unmanned system for multimodal environmental perception

Designed by Adam Lachman, Angel Pulido, and Ying Gao

Introduction

Purpose

To design and build an autonomous terrestrial rover capable of navigating an agricultural environment in order to locate, image, and quantify leaf chlorosis (yellowing) in plants

Scope

- **Computation**: Raspberry Pi 5 running ROS 2 Jazzy Jalisco for navigation.
- **Communication**: Dual ESP32 LoRa modules.
- **Sensors**:
  - dToF LiDAR
  - 1080p 30fps image/video camera
  - Voltage divider for battery monitoring
- **Analysis**: Leaf chlorosis (yellowing)
- **Interface**:
  - Rover: Python script managing state switching
  - Base Station: Gamepad input, map visualization and serial monitor output

Application Area

Project implementation is meant to be performed in an agricultural environment. Consent has been given for testing in

- R'Garden
- Agricultural Operations Fields

Solution Approach

Requirements

Components:

- 4WD chassis w/ 4× 3-6V DC gear motors
- 1x L298N Dual-Channel Motor Drivers
- 1× 11.1V 2.2Ah LiPo
- 1× 5V 5A Buck Converter
- PWM Limiter (Software)
- Voltage Divider circuit
- 1x Raspberry Pi 5
- 2x ESP32 LoRa modules
- 1x dToF LiDAR sensor
- 1× 1080p/30fps image/video sensor
- 2-Axis Pan/Tilt Servo arm
- Ros 2 Jazzy Jalisco
  - SLAM
  - Nav2

- **Hybrid Navigation**
  - LiDAR (global positioning)
  - YOLO (local vision)
- **Stop-and-Go Analysis**
  - Rover must detect a plant, stop, capture image, then resume movement
- **Offline Operation**
  - The rover must function without internet connection (after setup)
- **Telemetry**
  - Broadcasting of battery voltage, chlorosis %, and status to Base Station
- **Emergency Stop**
  - Must halt movement if LoRa connection lost or “STOP” command received

Challenges

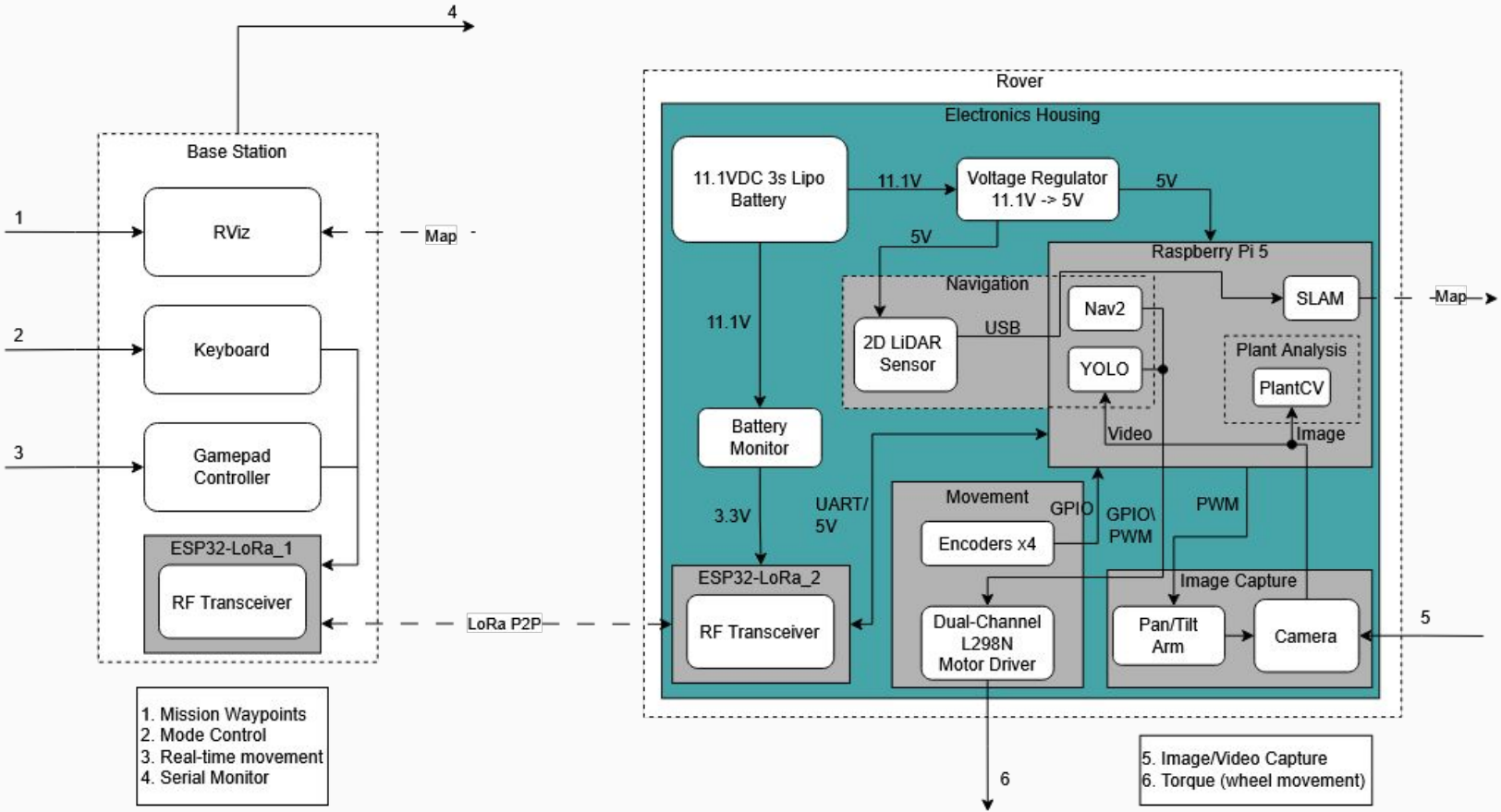
Main challenges:

- System synergy
  - RPI 5
  - Wheel-motor functionality
  - YOLO/PlantCV state switch
  - LoRa command/message broadcasting
- Circuit topology / Battery Load

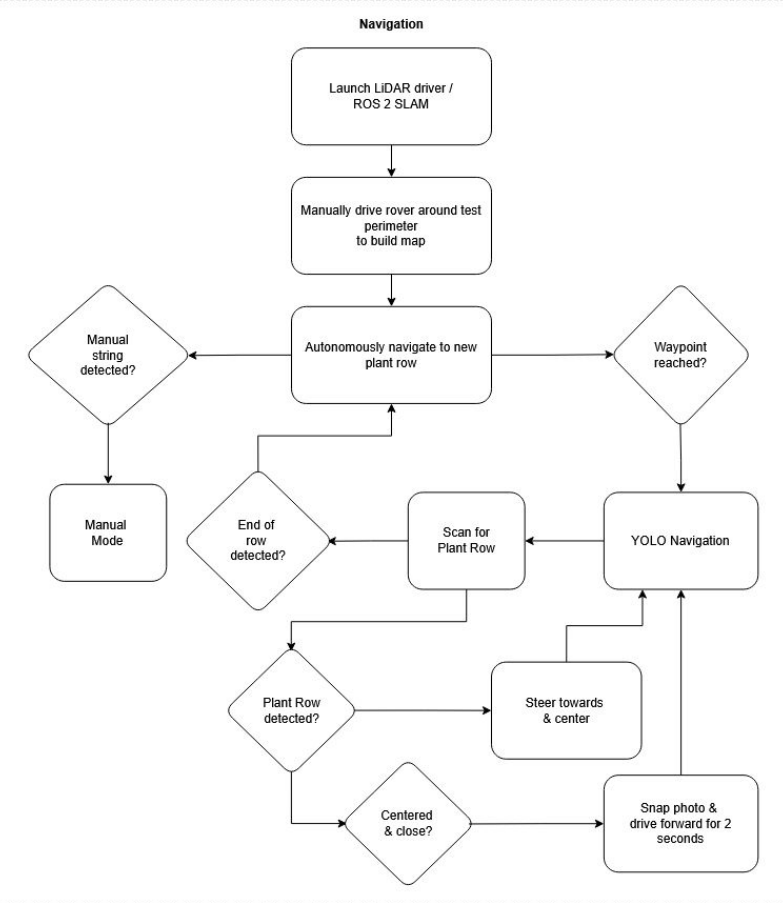
System Overview

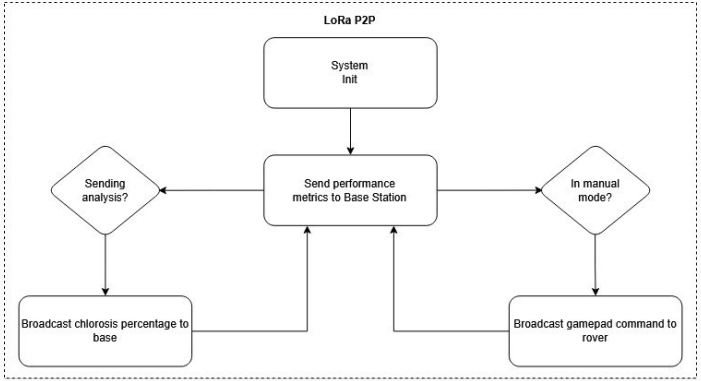
1. Power Compliance
   - Lane 1 (Regulated)
     - Pi 5, LiDAR, servos, LoRa
   - Lane 2 (Raw)
     - L298N driver
2. Hybrid Navigation
   - Manual SLAM mapping
   - Finite State Machine
3. Offline Telemetry
   - Proper handling of JSON data packets over 915MHz

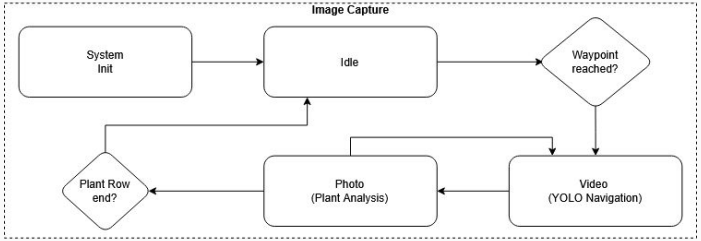
System Block Diagram

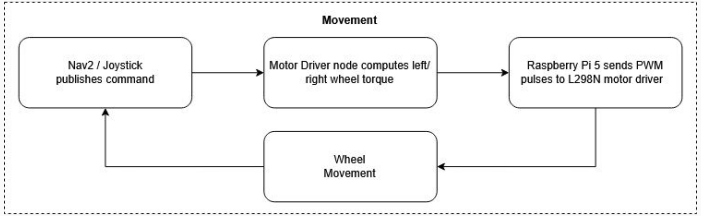


Subsystem Flow Charts



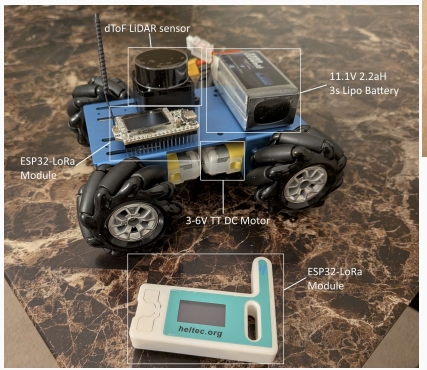


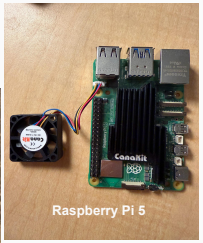


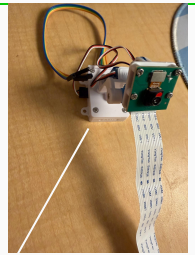


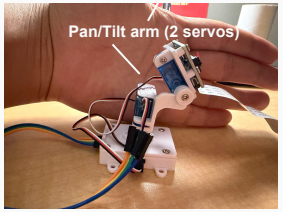
Implementation Plan

Hardware









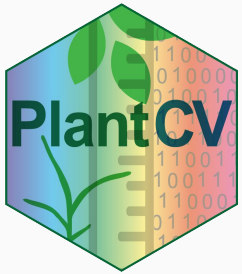
Hardware:

- **11.1V 2.2aH 3s LiPo battery**
- **ESP32-LoRa module x2**
- **dToF LiDAR sensor**
- **3-6V TT DC motor**
- **TB6612FNG motor driver**
- **Raspberry Pi 5**
- **Pan/Tilt arm (2 servos)**
- **Gamepad controller**

Rover Battery Draw:

- RPI 5 - 5V
- LiDAR - 5V
- ESP32 x2 - 3.3V

Software



Libraries/Algorithms:

- ROS 2 (Jazzy Jalisco)
  - Robot operating system / libraries
- YOLO
  - Plant/plant row detection
- PlantCV
  - Chlorosis analysis

UI design:

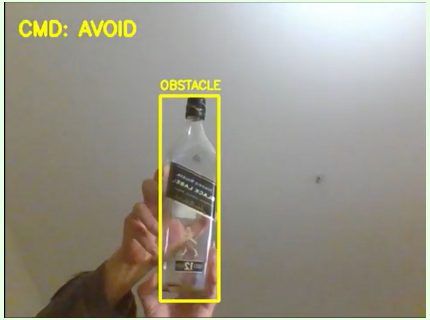
- RViz ROS sensor data visualizer

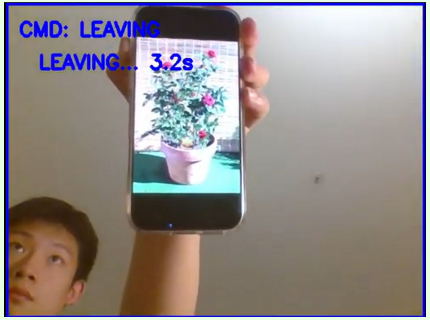
Data Collection/Analysis:

- Boustrophedon path navigation
- Stop-and-go phenotyping

Preliminary Results

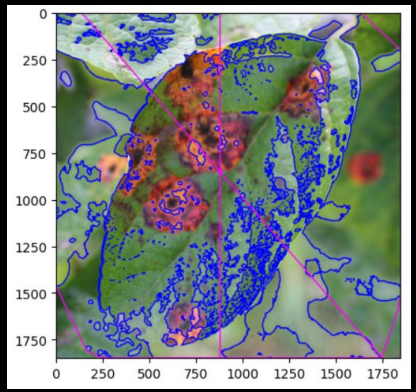
YOLO Object Detection





- Implemented control logic: "Stop & Snapshot" for flowers and "Avoid" for obstacles.

PlantCV Segmentation



Verification/Validation Plan

Performance Metrics

1. **Navigation**
   a. Cross-Track Error (XTE)
      i. Perpendicular distance between rover's center and ideal path.
   b. Human-Assist Rate
   c. Localization Drift
      i. Difference between where rover thinks it is and where it actually is.
2. **Machine Learning**
   a. YOLO
      i. FPS of the RPI 5
      ii. False positive/negative rate.
   b. PlantCV
      i. Percentage of unacceptable photos.
3. **Communication** (LoRa)
   a. Packet Loss Rate (PLR)
      i. Percentage of messages not sent.
   b. Signal Margin
      i. Available noise tolerance before signal loss.
4. **System Efficiency**
   a. Plants Analyzed per hour
   b. Energy cost per plant

Expected Results

- A photo of a plant leaf with chlorosis analysis results.
  - E.g. 'Chlorosis: 30%'
- Mission performance metrics such as total plants scanned, runtime, etc.

Testing Procedures

- Battery Load (Discharge) Test
  - Discharge Capacity
- Mobility Testing
  - Simulated terrain environment
- Deployment Testing
  - Complete system trial

Project Management

Chassis Assembly	All	Complete	• Assembled metal frame, 3-6V TT DC motors, and omni-directional wheels.
ROS 2 Installation	Angel	Complete	• Installed ROS 2 Jazzy Jalisco onto the Raspberry Pi 5 and to desktop.
YOLO / Camera	Ying	Status	• Deployed YOLOv8n model with OpenCV for real-time visual perception. • Implemented control logic: "Stop & Snapshot" for flowers and "Avoid" for obstacles.
PlantCV / LiDAR	Adam	Status	• Plant segmentation using binary mask • LiDAR to be implemented
LoRA / Motor Driver	Angel	Status	• LoRA modules configured for P2P communication. • L298N motor driver
Circuit Topology / Layout	All	Status	• Preliminary system design and component acquisition complete. • Physical component placement to be determined.

Thank you.

Unmanned System for Multimodal Environmental Perception

EE175A Section 22 Group 24
Adam Lachman, Angel Pulido, Ying Gao

Introduction

purpose, scope, application area/use case

Solution Approach

- What are requirements and challenges

- Overview of your system; how your system design meets the requirements

System Block Diagram

Subsystem Flow Charts

Implementation Plan

- Hardware (what hardware parts to use; specs related to the system requirements)

- Software (libraries/algorithms to use; UI design; data collection/analysis methods type, etc)

- Show if you have preliminary results

Validation/Verification Plan

metrics to assess performance, expected results, brief testing procedures

Project Management

Gantt chart, work distribution, current progress