Retriever Rover

Proposal Presentation

Team C0: Matthew Mao, Jasmine Chao, Joonyoung Huh

Users

The elderly and people with stiff joints may have a hard time bending down. According to pmc, nearly a quarter of adults experience trouble bending down or kneeling. Finding objects on the floor can be a painful and risky process.

Use Case

Retriever Rover is a robot that can

1. Receive a command to start searching through an app
2. Find user-specified objects: (start with **wallet**, extend to **keys**, **pill bottle, and phones**)
3. Delivers objects via a ramp, allowing users to avoid bending down
4. Provides status updates for user through text/email

Image created using Generative AI

ECE Areas: Software Systems, Signals and Systems

Use Case-Requirements

Be able to find the object (if it is on the floor) **95%** of the time

Grip the object (if found) **90%** of the time

Carry object up the ramp **95%** of the time

Once user submits a request, have the robot start searching within a reasonable delay of **3s**

Notify the user through text/email **100%** of the time and within a reasonable delay (**3s**) when it

- finds the object
- drives up a ramp to an elevated space that is accessible to user

Technical Challenges

Gripping

Challenge:
- Grabbing objects with 90% success

Mitigation:
- Depth Camera
- Testing and Calibrating

Vision

Challenge:
- Identifying and finding objects with 95% success

Mitigation:
- YOLOv8
- Custom Dataset

Navigation

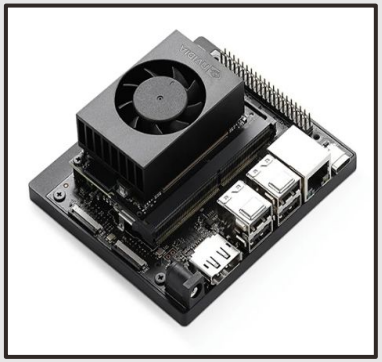
Challenge:
- Avoiding obstacles and travelling up ramp

Mitigation:
- Nav2
- Only indoors

Solution Approach - Robot

Hardware

- Jetson Orin Nano

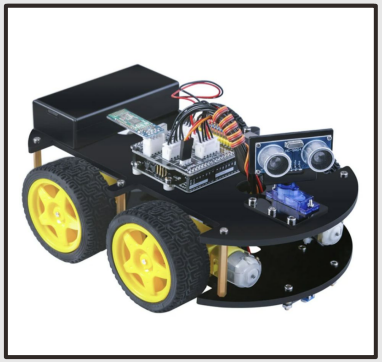


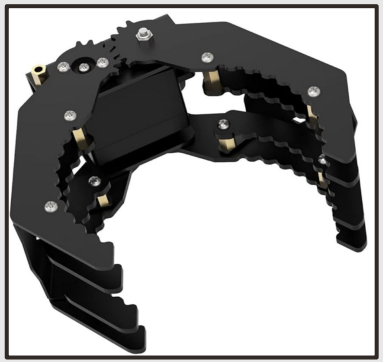
Camera

- RealSense D435 Depth Camera

Base

- Elegoo Car kit, or similar

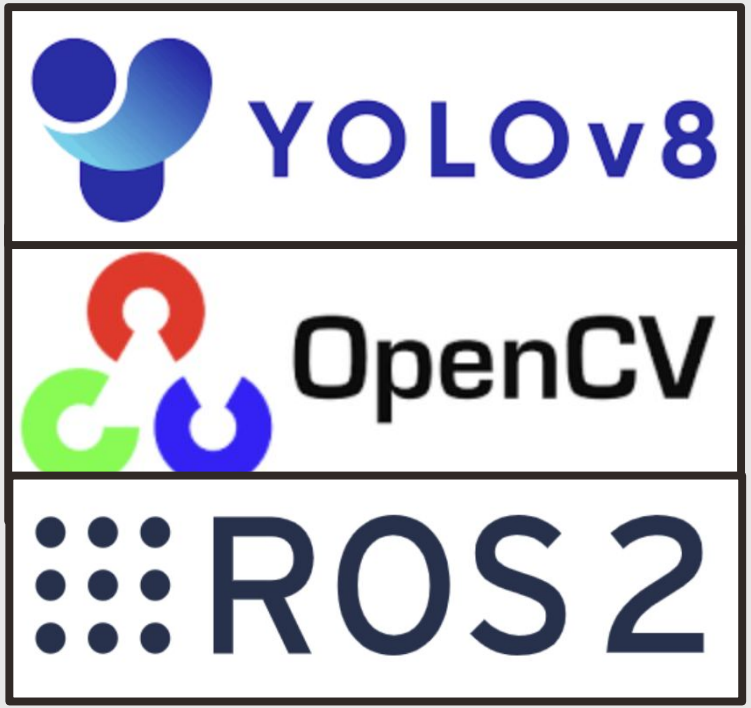




Gripper

- Servo based Claw

Solution Approach - Software



Software

- YOLOv8
  - Computer vision and classification
- OpenCV
  - Live video inference
- ROS2
  - Robot operating system
- Nav2
  - Navigation framework for ROS2
- WebApp
  - Hosted on AWS
  - REACT

Testing Verification and Metrics -Robot

Requirements	Testing	Expectation Metrics
Object detection	Place target object at random floor locations and run detection over 100 trials using live camera feed	≥ 95% detection accuracy
Object Grasping	Perform 100 repeated pickup trials for each object type (keys, wallet, pill case)	≥ 90% successful grasps without object damage
Navigation to and up the object/ramp (delivery)	Allow the robot to pick up an object and attempt to travel to and up the ramp. Perform 100 trials, and each trial starts when the robot has successfully picked up an object	≥ 95% successful navigation without dropping the object, or flipping over

Testing Verification and Metrics - Web App

Requirements	Testing	Expectation Metrics
User Notification	During our trials for detection and delivery, we will also keep track of whether or not notifications are being sent.	100% successful notification delivery
Low Latency (App ↔ Robot)	For 20 trials, we will measure the time from user submitting target via web app to robot initiating motion, and from task completion to user receiving notification	Command latency ≤ 3 s Notification latency ≤ 3s

Tasks and Division of Labor

Matthew

Robot mechanisms
- Attaching the gripper to the drivetrain
- Modifying the gripper to be compatible with objects

Notification feature
- Getting robot to send a signal to app that object has been found
- Getting app to send a notification to user

Jasmine

Robot control
- Firmware for drivetrain and gripper
- Search algorithm for object

Computer vision integration
- Uploading YOLO on Jetson

Ramp construction
- Construct a ramp for the robot that is tall enough for users to access without bending down

Joon

Robot prediction
- Using YOLO model on Jetson to make predictions to detect object

App development
- A form for user to upload the target object
- Getting app to send target to robot

Schedule

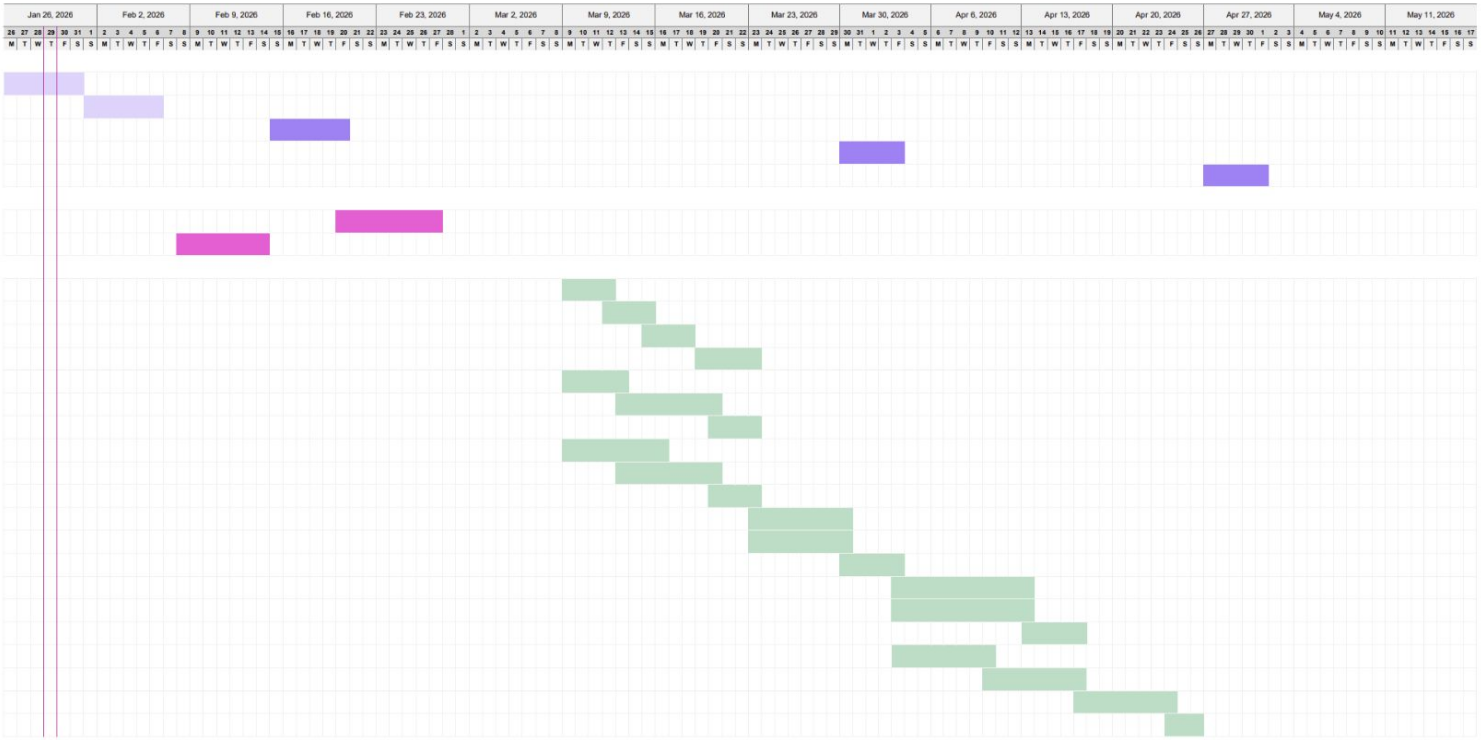
Retriever Rover

Team C0

Project start: Fri, 1/30/2026

Display week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Logistics				
Website Setup	All	100%	1/19/26	1/31/26
Proposal Presen	All	100%	2/1/26	2/6/26
Design Presenta	All	0%	2/15/26	2/20/26
Interim Demo	All	0%	3/30/26	4/3/26
Final	All	0%	4/27/26	5/1/26
Project Planning				
Order Materials	All	0%	2/20/26	2/27/26
Finalize Design	All	0%	2/8/26	2/14/26
Project				
Train YOLOv8 or	Joon	0%	3/9/26	3/12/26
Train YOLOv8 or	Joon	0%	3/12/26	3/15/26
Train YOLOv8 or	Joon	0%	3/15/26	3/18/26
Slack	Joon	0%	3/19/26	3/23/26
Assemble Robot	Matthew	0%	3/9/26	3/13/26
Make grippers w	Matthew	0%	3/13/26	3/20/26
Slack	Matthew	0%	3/20/26	3/23/26
Setup ROS2	Jasmine	0%	3/9/26	3/16/26
Setup Nav2	Jasmine	0%	3/13/26	3/20/26
Slack	Jasmine	0%	3/20/26	3/23/26
OpenCV with YO	Jasmine and Joon	0%	3/23/26	3/30/26
Test and refine c	Matthew	0%	3/23/26	3/30/26
Slack	All	0%	3/30/26	4/3/26
Make app	Joon	0%	4/3/26	4/13/26
Make notifcaiton	Matthew	0%	4/3/26	4/13/26
Combine app + r	Matthew and Joon	0%	4/13/26	4/17/26
Construct Ramp	Jasmine	0%	4/3/26	4/10/26
Slack	Jasmine	0%	4/10/26	4/17/26
Testing/Debuggi	All	0%	4/17/26	4/24/26
Slack	All	0%	4/24/26	4/26/26



Minimum Viable Product

Our robot must autonomously detect, retrieve, and deliver **one** user-specified object (**wallet**) from the ground to a user-accessible height by climbing a ramp, without human intervention, achieving at least 80–90% success across repeated trials.

Stretch Goal

1. Extend object detection to **multiple** household items (**keys, phones, pill bottles**)
2. Replace fixed-height delivery (ramp) with an elevator mechanism that allows users to select delivery height via the app