SDP25

Team #15
Workspace Wizard

Meet the Team

Aryaman Ghura
CompE & CS

Algorithms, Integration, and Optimization

Ishaan Salian
CompE

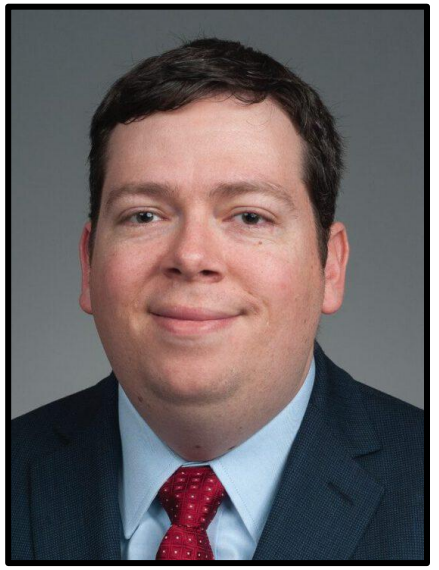
Mechatronics & PCB
Logistics

Mary Esenther
CompE

Power Delivery & Camera Processing

Kavya Manchanda
CompE

Model Training
Budget



Professor Marco Duarte

Advisor

Problem Statement

- A cluttered desk can negatively affect productivity and mental health.
- Studies show that messy workspaces increase stress. [1]
- 57% of Americans associate messy desks with laziness. [2]

[1] "The Psychological Consequences of Clutter," Psychology Today, 2021.
[2] L. Alton, "The Negative Relationship Between a Messy Desk and Productivity," Inc.com, Feb. 16, 2017.

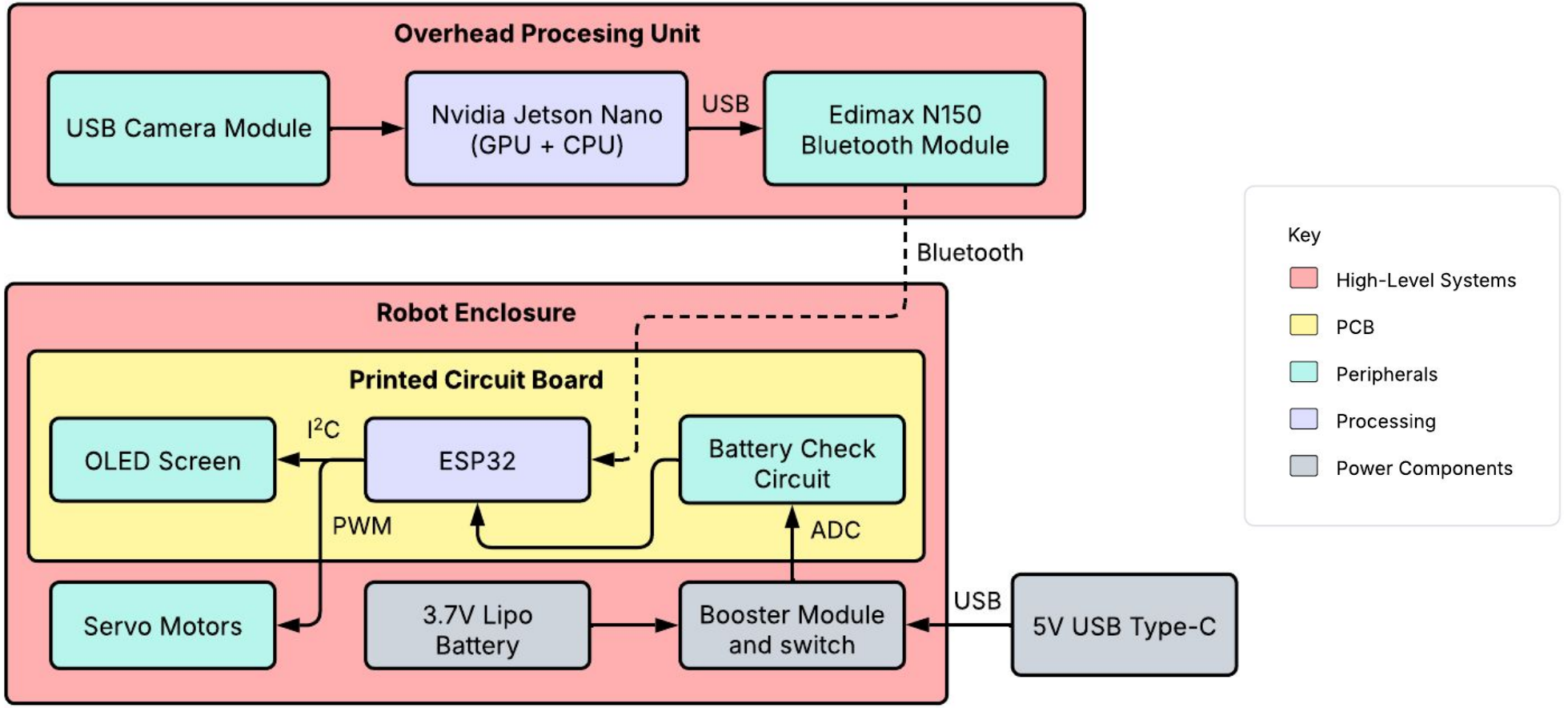
Project Goal

To create a distributed system that can organize your workspace autonomously by employing object detection and path finding.

Project Testing Plan based on Design Specifications (Updated)

Design Specification	Testing Plans
Detect human absence	Robot movement starts only after 2 minutes of human absence in at least 98 out of 100 trials (reduced from 20 minutes for demo purposes)
Map viable paths for object placement	The robot should successfully navigate around obstacles to find viable paths without getting stuck, if there exists one
Orient Objects	Orient objects such that their bounding boxes consume the least area
Identify objects to avoid versus knoll	System should identify various forms of objects (open cup, bottle, laptops) with correct classifications in at least 98 out of 100 trials
Recognize table boundaries	If either one of the ArUco markers is missing, or the robot is not within the space defined by the markers, the program will identify this and tell the user
Run for 30±5 minutes on one full charge	Run robot for 30 minutes while monitoring battery levels
Knoll up to 10 (varying) objects	The robot should knoll all 10 objects accurately in at least 95% of trials, if knolling path exists

Hardware Block Diagram (FPR)



Software Diagram (MDR)

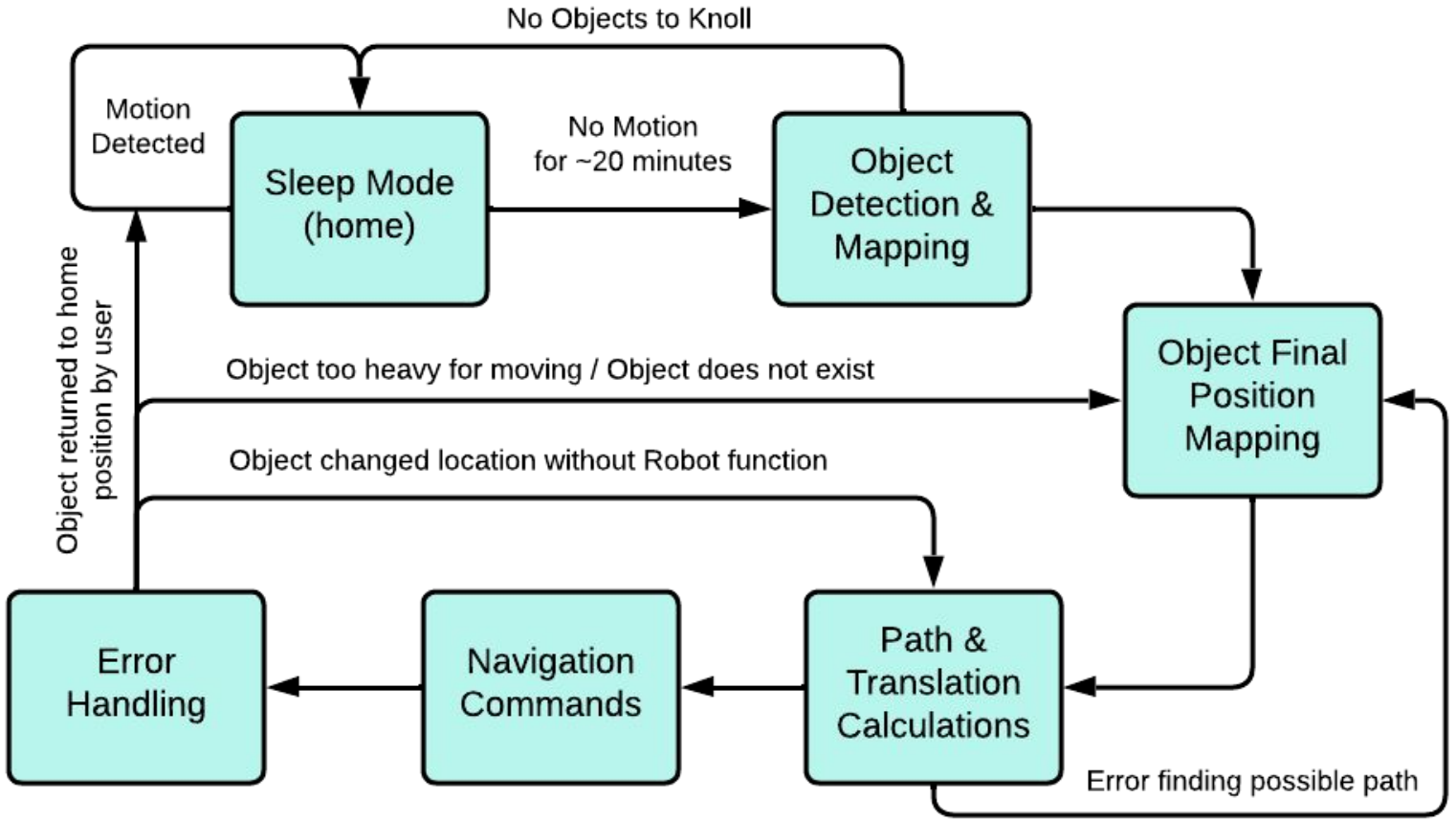
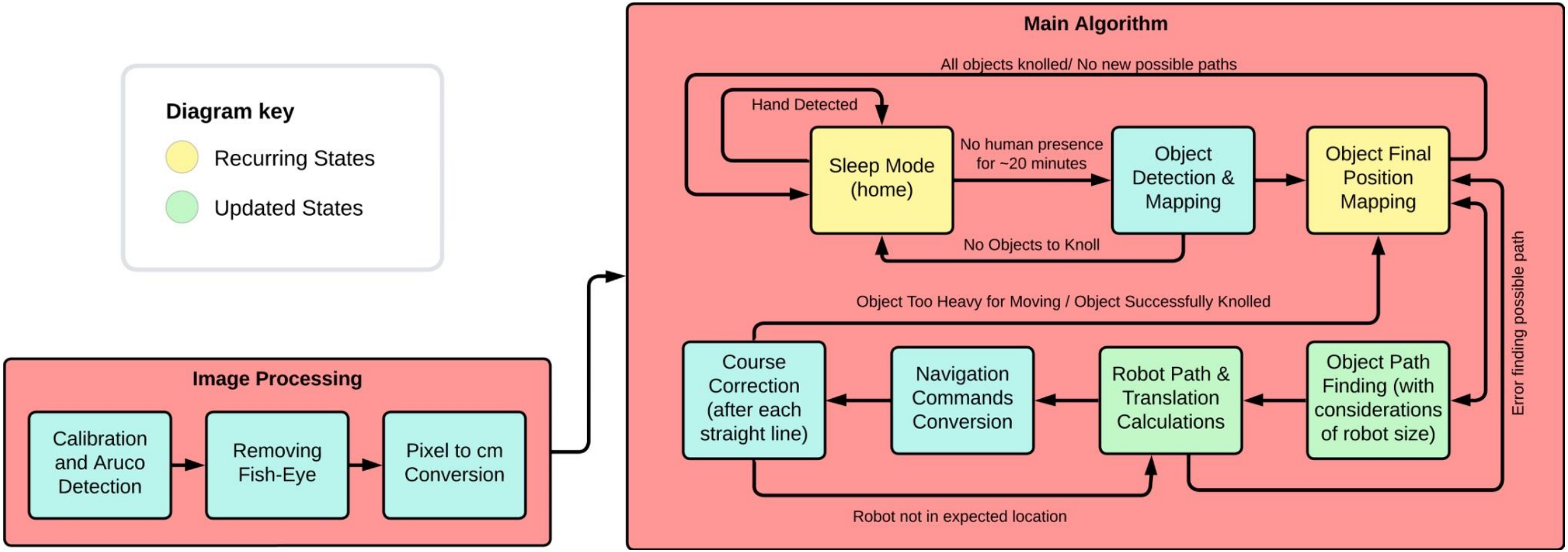


Image Processing Libraries:
OpenCV
Media Pipe, DarkNet

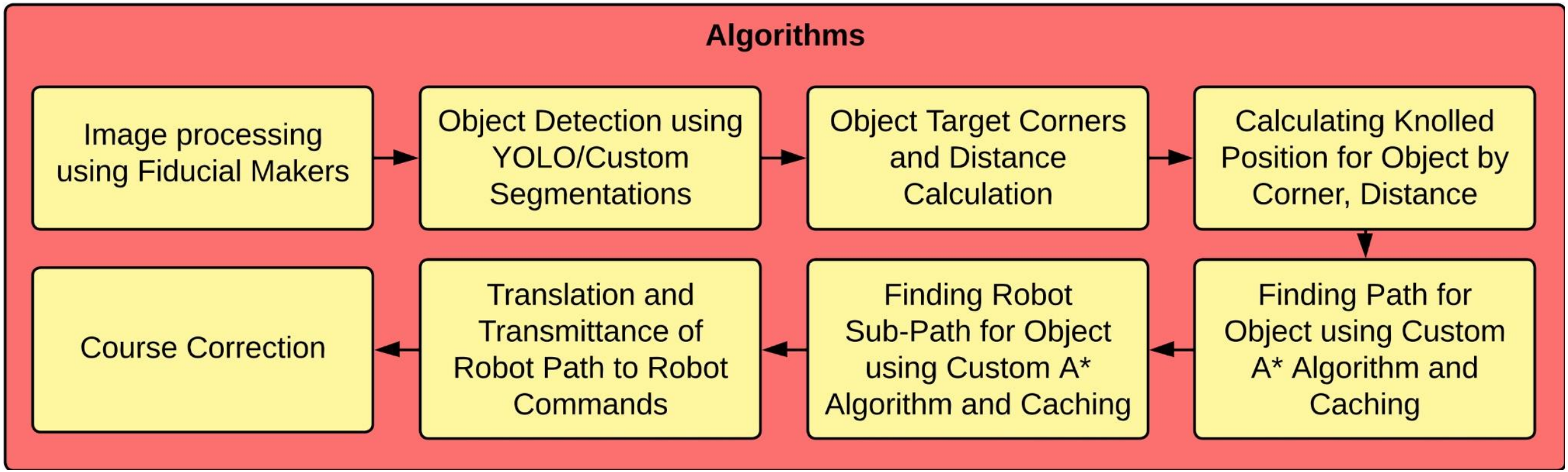
Basic Libraries:
Heap, Numpy, Math, Copy, Time, os
Networkx

BLE Communication Libraries
asyncio, re, bleak

Updated Software Diagram (FPR)



Algorithms Flowchart



Integration Progress

- Nvidia Jetson integrated with entire system
  - All software libraries successfully integrated into Jetson
  - Bluetooth module now successfully communicating with ESP32
- Software Integration
  - Aruco Markers, Knolling, Path Planning, and Robot communication now successfully integrated and working efficiently

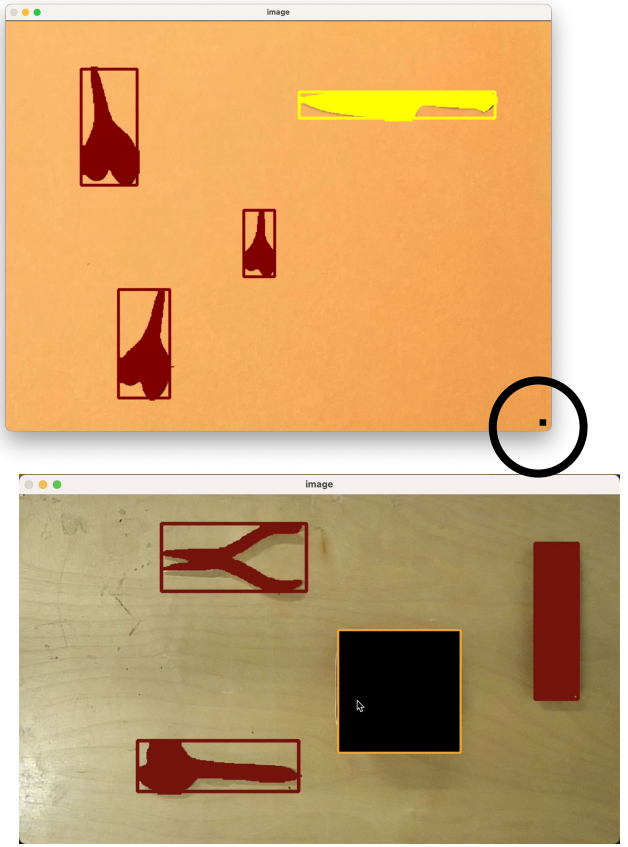
Optimization Progress

- Aruco Marker Detection Optimized
  - Only two markers now required to make the final solution less invasive of user workspace
  - Markers can be placed in any orientation (automatically detects required corners)
- Pixel to Centimeter Conversion Optimized
  - Distance between edges of Aruco used to calculate conversion ratio (as opposed to calculating distance between corner Arucos)
- Path Finding (next slide)

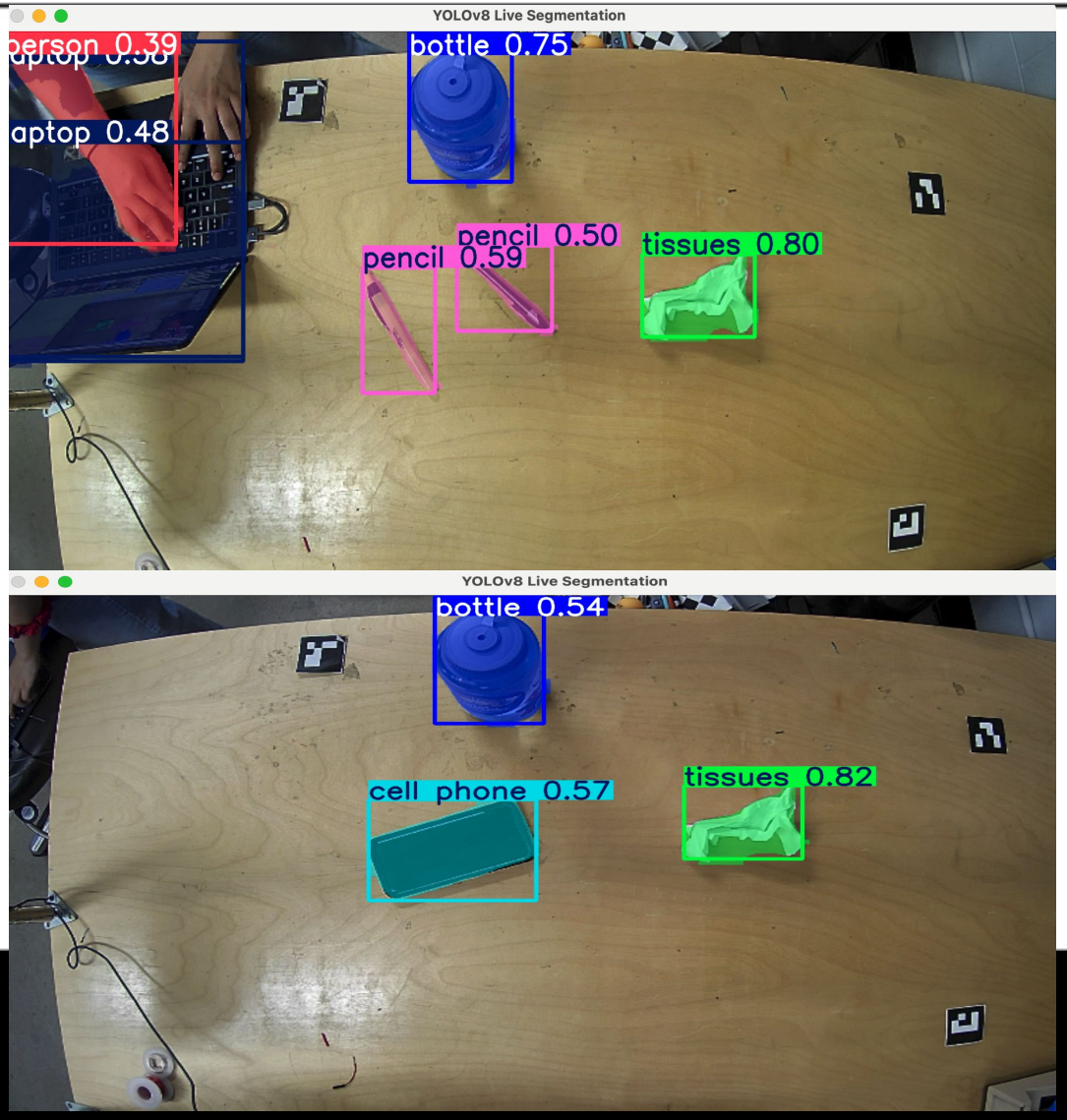
Path Finding Optimization

- Path Finding complexity extensively reduced
  - ~15 minutes for 5x5 pixel robot box to ~20 seconds for 170x170 pixel robot
  - Efficiently works on Jetson

- Attempts:
  - Greedy Algorithms and Weighted A*
  - Reducing Grid by diminishing Pixel Resolution
  - Separating Object Path Finding and Robot Path Finding
  - Removing dependency on networkx library to make grid
  - Caching robot positions when finding valid object path
  - Employing numpy algorithms for array operations
  - Precalculating viable locations for robot when finding sub-path



Custom Trained Model



Objects we are identifying:

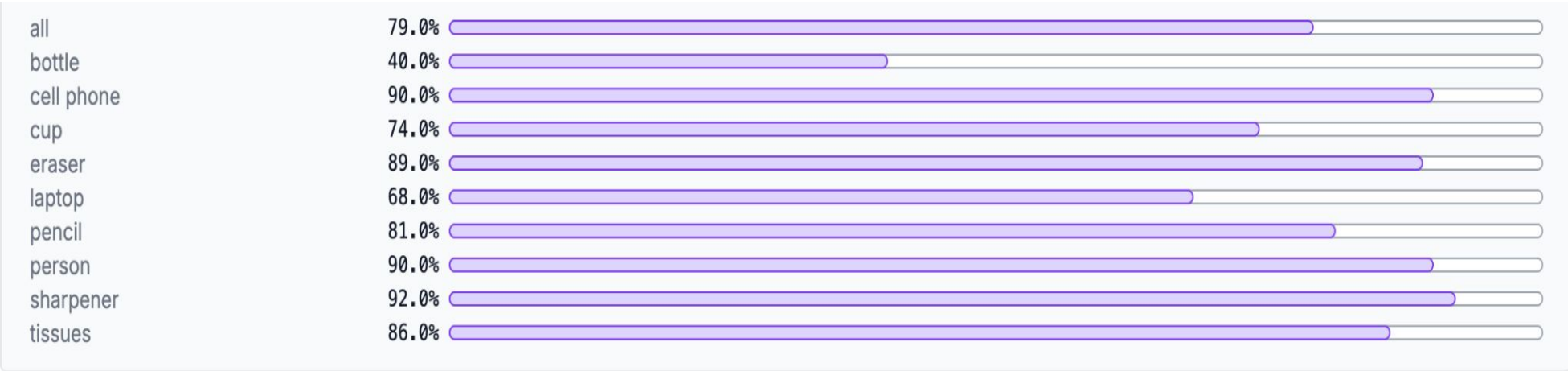
- Pencil
- Bottle
- Cup (avoid)
- Eraser
- Sharpener
- Bottle (avoid)
- Cell Phone
- Laptop (avoid)

Image sources: COCO instance segmentation 2017 dataset, self-captured images, Roboflow Universe models, GettyImages, Canva

- Trained dataset with Ultralytics Python library and YOLO API
- Used Roboflow for labeling & modifying dataset and downloading data.yaml file
- Trained instance segmentation model on pre-trained YOLOv8m-seg model

Merging Models

- Merged YOLO categories of bottle, cup, cell phone, laptop, person with custom categories of pencil, sharpener, eraser, tissue.
- Step 1: Download COCO dataset and annotations in JSON format
- Step 2: Parse the JSON file to filter annotations and filenames
- Step 3: Convert these annotations into normalized labels: ratio of coordinates and image size and width/height and image size
- Step 4: Filter images from the dataset to match the images in the labels.txt file
- Step 5: Merge datasets and create a new data.yaml file with all class names



Accuracy:

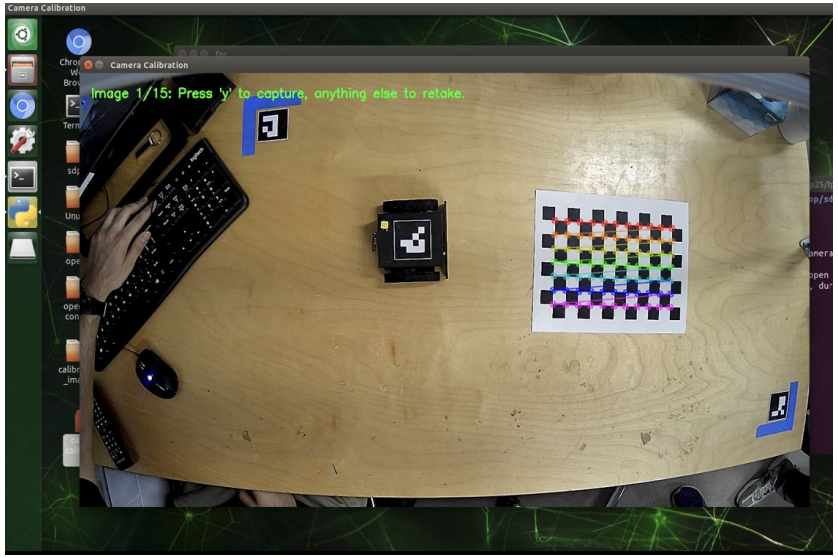
- We were able to achieve an accuracy that does not interfere with the way we treat objects or other specifications.
- Accuracy tends to be in the higher range of over 70%, matching YOLO's results.

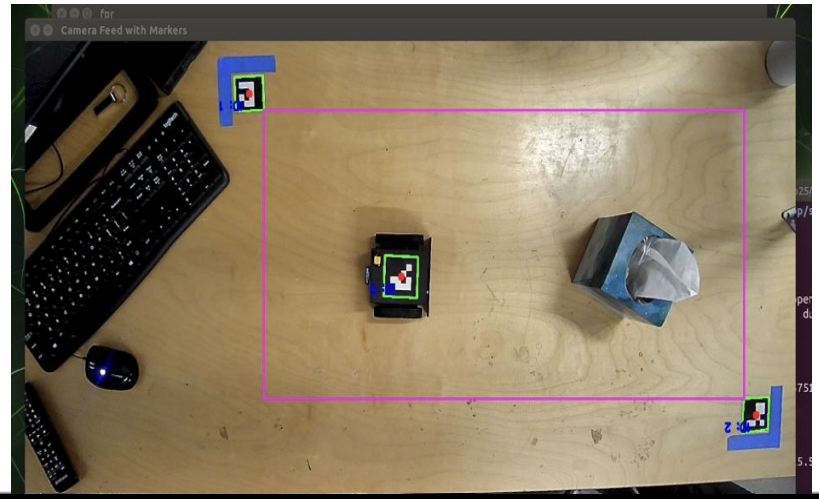
Fiducial Marker Tracking

Step 1: Calibrate camera with chessboard images

Step 2: Crop image using outside fiducials and remove “fisheye” effect

Step 3: Print coordinates of the robot fiducial marker’s corners (marker 0)





FPR Changes:

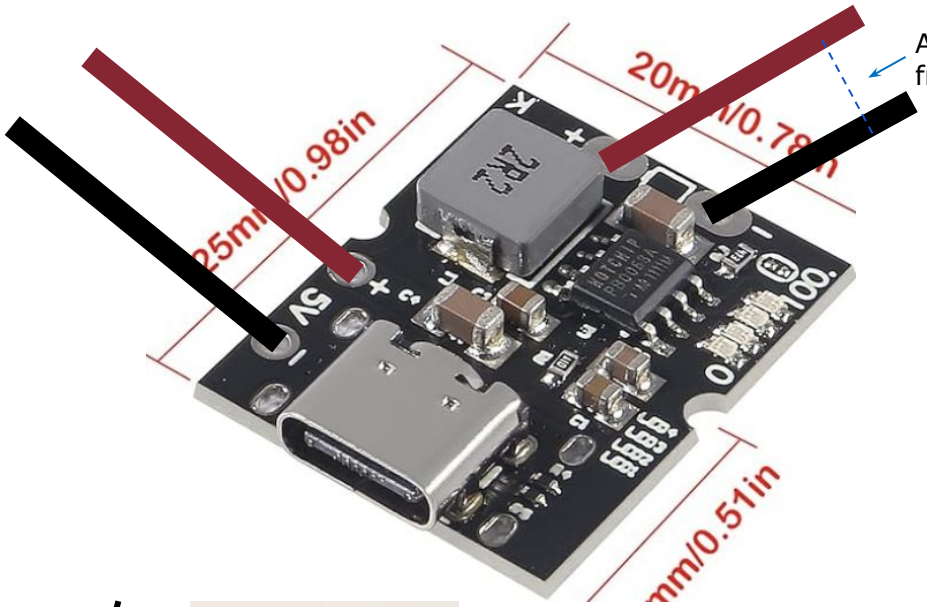
- Only 2 markers rather than 4
- Tracks pixel to cm ratio using marker size so that movement commands can be generated based on robot location

Power System (same as CDR)

ESP32/PCB
(250mA max)
Motor 1
(600mA max)
Motor 2
(600mA max)

= **1.45A max**

Limit: 2A



Voltage Tracking:
Battery terminals feed into ADC pin of ESP32 on the PCB

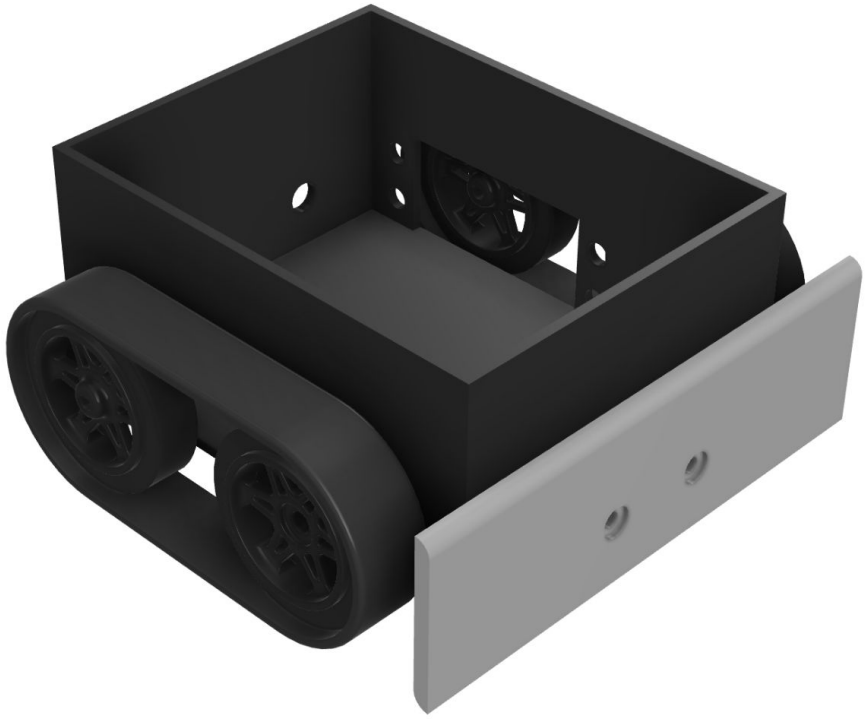
3.7V 5000mAh LiPo
Time = Capacity/Current

Expected:
5Ah/1.45A = **3.45h**
(actual: limited by boost)

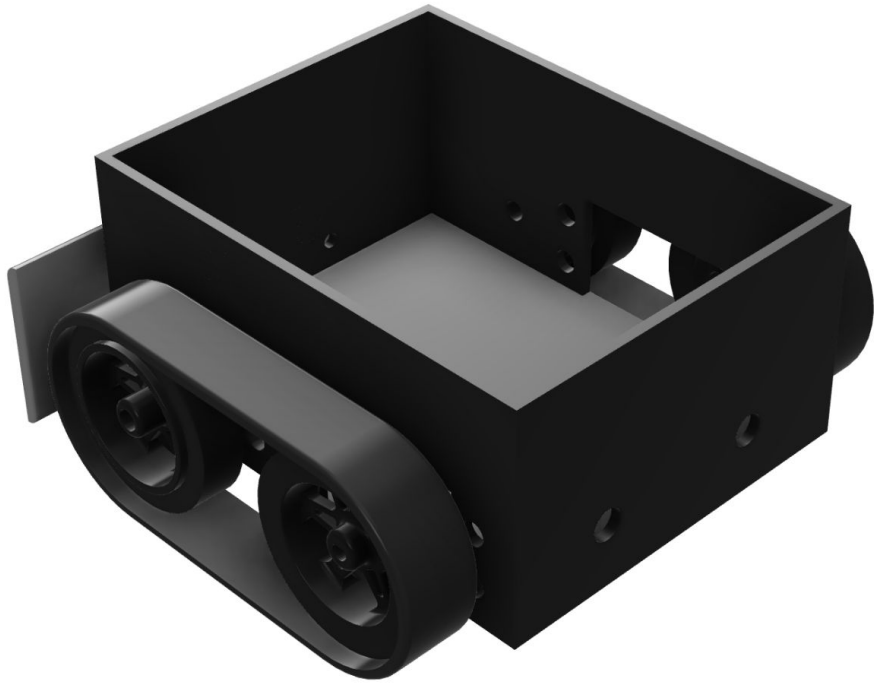
Charging:
5Ah/(charge current)

Expected charging:
5Ah/3A = **1.67h**

Robot Design (CDR)



Front



Back

Shortcomings:

- No charger
- Top did not fit
- No defined place for ArUco marker
- No place for display

Current Robot Design (FPR)

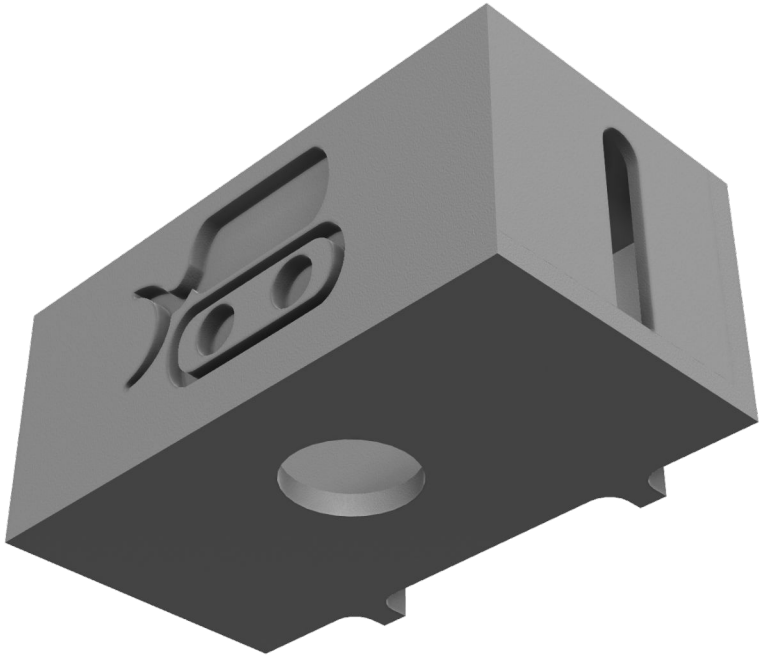
Front

Back

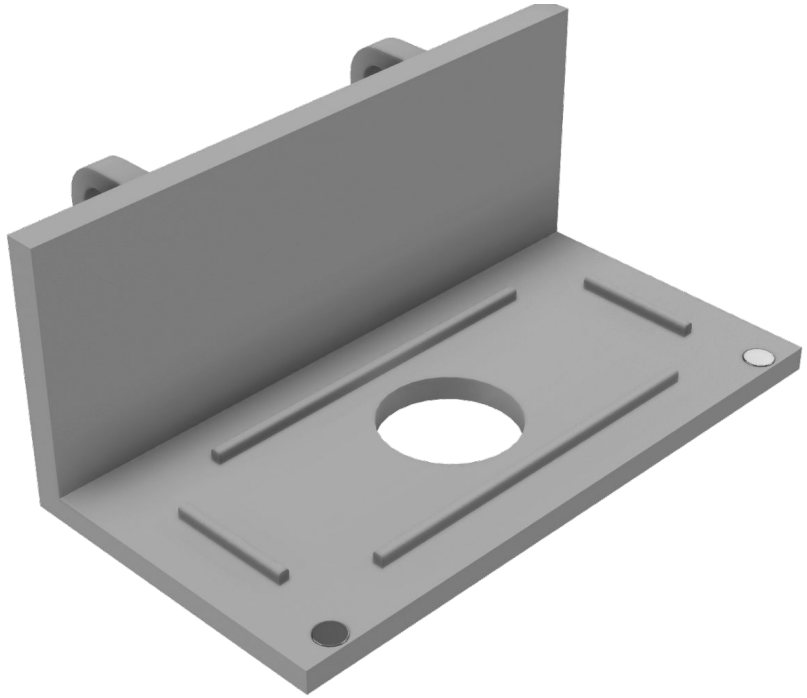
Updates:

- Added BMS mounting and Type-C port
- Top snaps into place with magnets
- Added On-OFF button to top
- Added inlay for ArUco

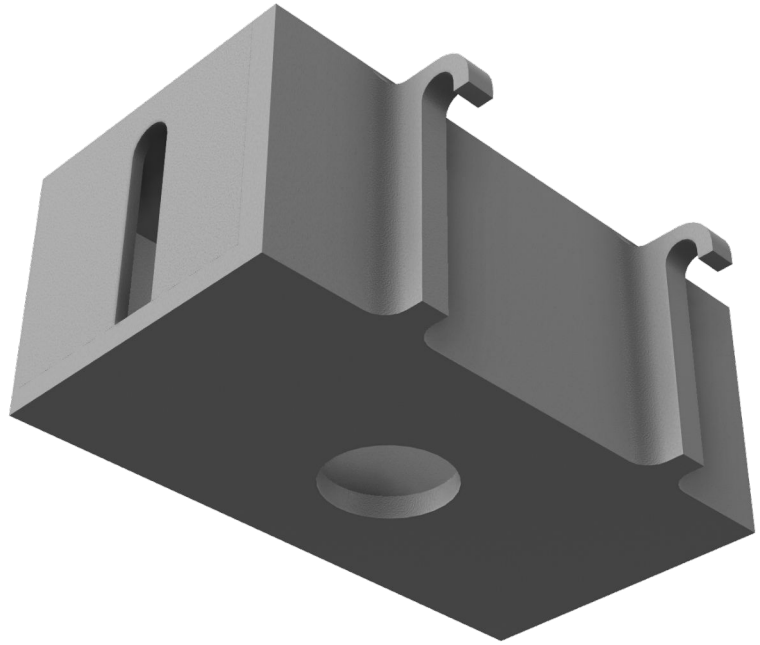
Overhead Camera Enclosure (FPR)



Front-Bottom



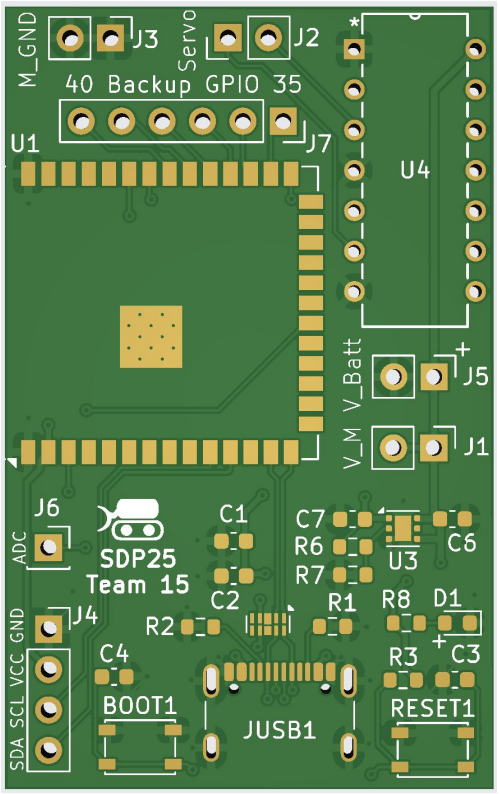
Front-Top



Back-Bottom

PCB Plan (CDR)

Populated PCB

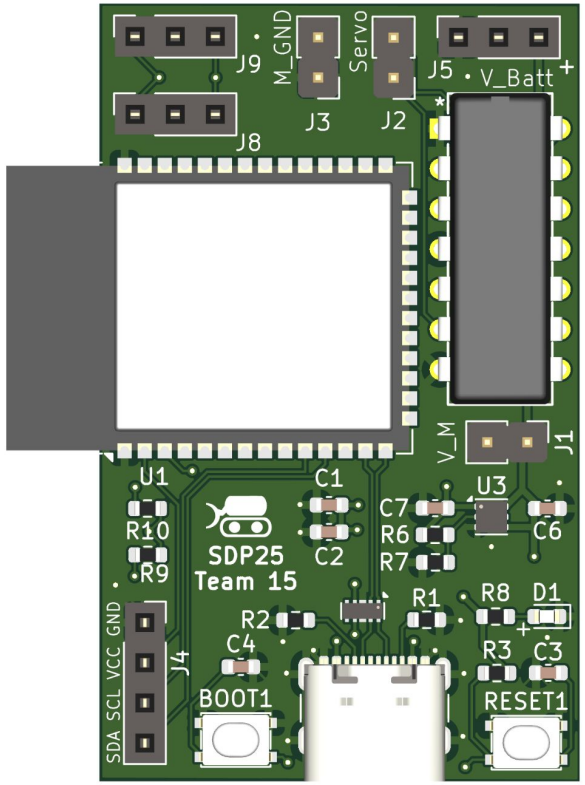


Unpopulated PCB

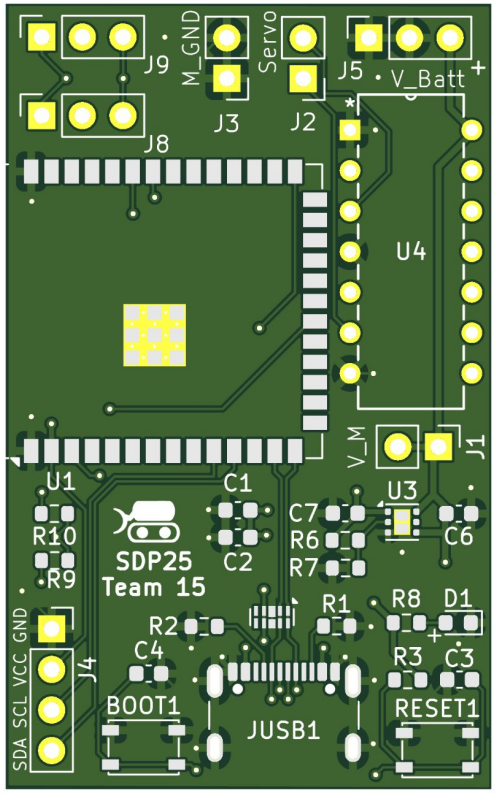
Shortcomings:

- Backup pins not needed
- Battery % circuit needed
- Traces are not optimally placed for proper wiring inside robot

Updated PCB (FPR)



Populated PCB



Unpopulated PCB

Updates:

- Removed backup pins.
- Added battery % reading circuit on board.
- Slightly smaller footprint.
- Improved trace placement for clean wiring inside robot.

CDR Deliverables (Remaining)

Overhead Computer + Software

- ✔ Integration of all software systems with one another and on the Nvidia Jetson. - Aryaman
- Demo of robot knolling three rectangular objects of weight below the robot's knolling threshold that are scattered across a study desk with ample space in between objects for robot to propel the object in any direction. The study desk will be initialized with a fourth object that must be ignored but considered during path planning. - Aryaman
- ✔ Finalized setup of the body of the camera on top of the desk. - Ishaan

Robot

- ✔ Robot (including revised PCB design) is able to navigate according to BLE commands from the Jetson. - Ishaan

FPR Deliverables

Overhead Computer + Software

- Meet any remaining CDR Deliverables - Team
- ✔ Optimize code for speed and efficiency - Aryaman
- ✔ Simplified fiducial tracking mechanism and conversion to movement - Aryaman, Mary
- ✔ Develop orientation algorithm - Aryaman
- ✔ Course correction - Aryaman, Mary
- ✔ Optimize the instance segmentation custom model for accuracy that meets the YOLO segmentation model - Kavya

Robot + Power Delivery

- ✔ Revised PCB design with integrated power management subsystem - Ishaan
- ✔ Improved robot and overhead camera enclosure - Ishaan
- ✔ Improved movement accuracy - Ishaan

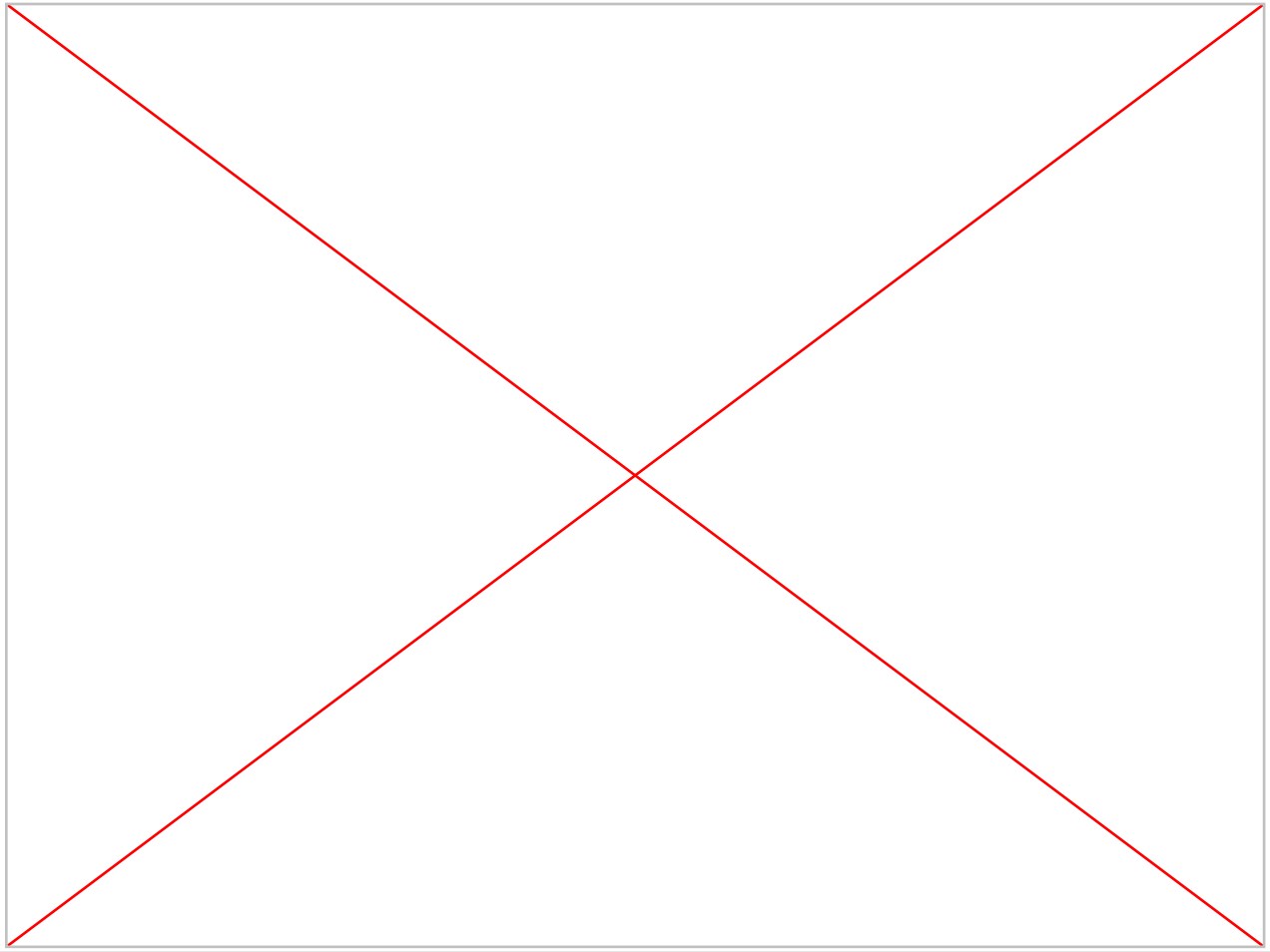
Project Expenditures (so far)

Tracks	$22.90	For the robot wheels	Legend:
PCB components - MDR	$5.03	Robot PCB Components	Overhead Computer
PCB print + shipping - MDR	$19.66	PCB print and shipping	Robot
NVIDIA Jetson Nano	**	Overhead Computer	PCB
MicroSD Card 64 GB	9.65	Overhead Computer storage	
Type-C USB 5V 2A Boost Converter Step-Up Power Module Lithium Battery Charging Protection Board LED Display	23.98	BMS for the robot	
3.7V Lipo Battery 5000mAh Rechargeable Lithium Polymer Battery Pack with JST PH2.0mm Connector for Electronic Device	28.98	Batteries for the robot	
PCB components - CDR	58.6	Robot PCB Components	
PCB print + shipping - CDR	42.26	PCB print and shipping	
Servo Motors (2)	**	Motors for movement of robot	
PCB components - FPR	29.9	Robot PCB Components	
PCB print + shipping - FPR	31.3	PCB print and shipping	
Total	$272.26		
Remaining	$227.74		

https://docs.google.com/spreadsheets/d/1IjUjBBgdBLAvl8chtcBwEpMHnhw8kTVnyTxhHfsBHh4/edit?gid=0#gid=0

** Acquired from M5

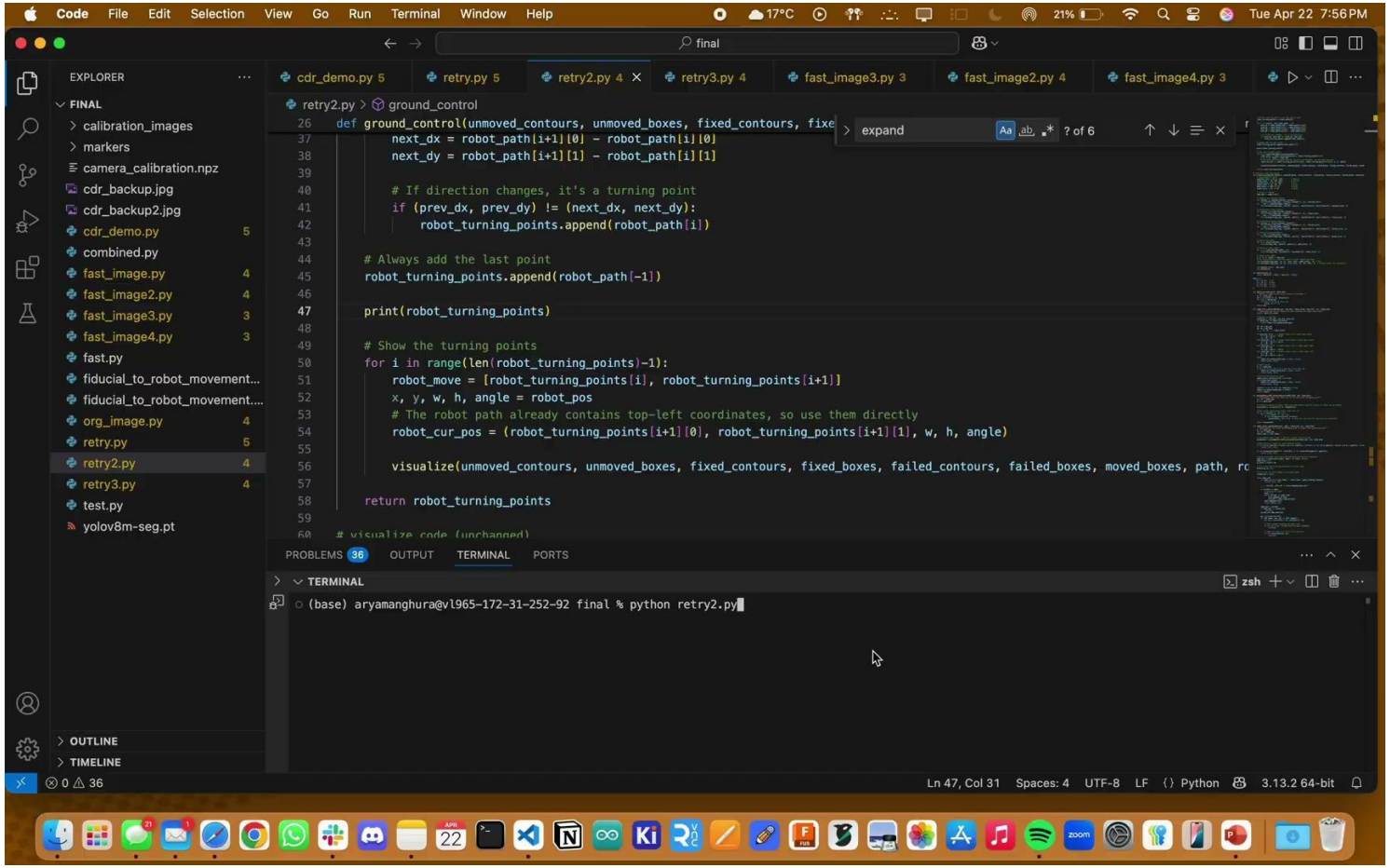
Design Specification 1 - Detect Human Absence



- Detect human absence for 2 minutes and return `True` once no “person” is found for 2 minutes.
- Runtime and “person” clock displayed at the top

* Please click on video preview to open link (for PDF)

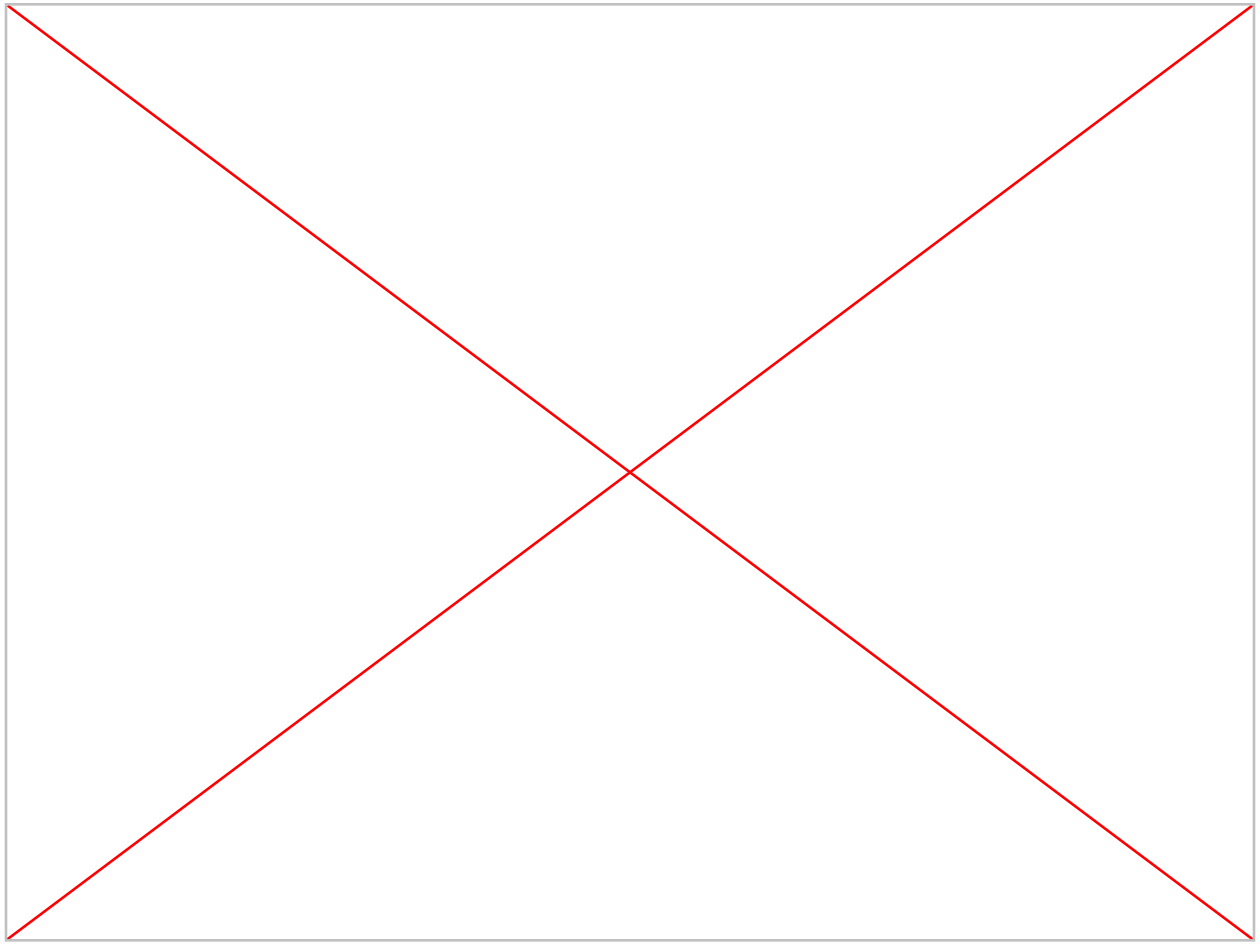
Design Specification 2 - Path Finding



- Three objects and the robot are detected on the table though live feed.
- A path is found for all three objects as it progresses one by one (red labelling for unmoved objects and green outline for final location)
- Blue line dictates object path, and the robot path is animated by the black box moving

* Please click on video preview to open link (for PDF)

Design Specification 3 - Orient Objects



- Robot knolling a remote in real time and orienting its edge with the table boundary.

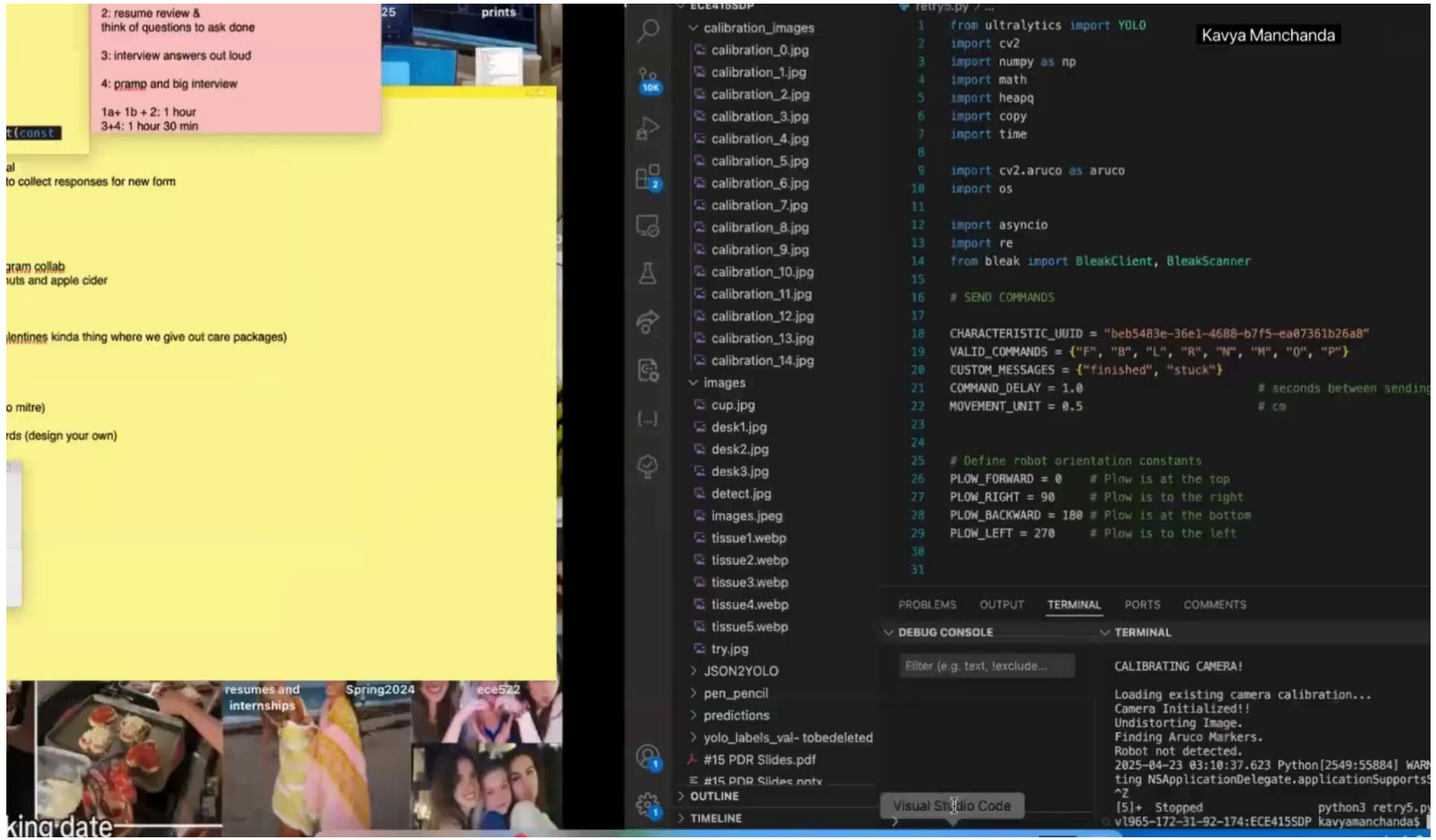
* Please click on video preview to open link (for PDF)

Design Specification 4 - Identify objects to avoid

- Objects to avoid are labeled in as: “person”, “bottle”, ”cup”, ”laptop”
- Smaller objects have lesser confidence when camera is placed far - this goes for both YOLO and custom model
- Sharpeners, eraser and pencils have high confidence when camera is placed closer

* Please click on video preview to open link (for PDF)

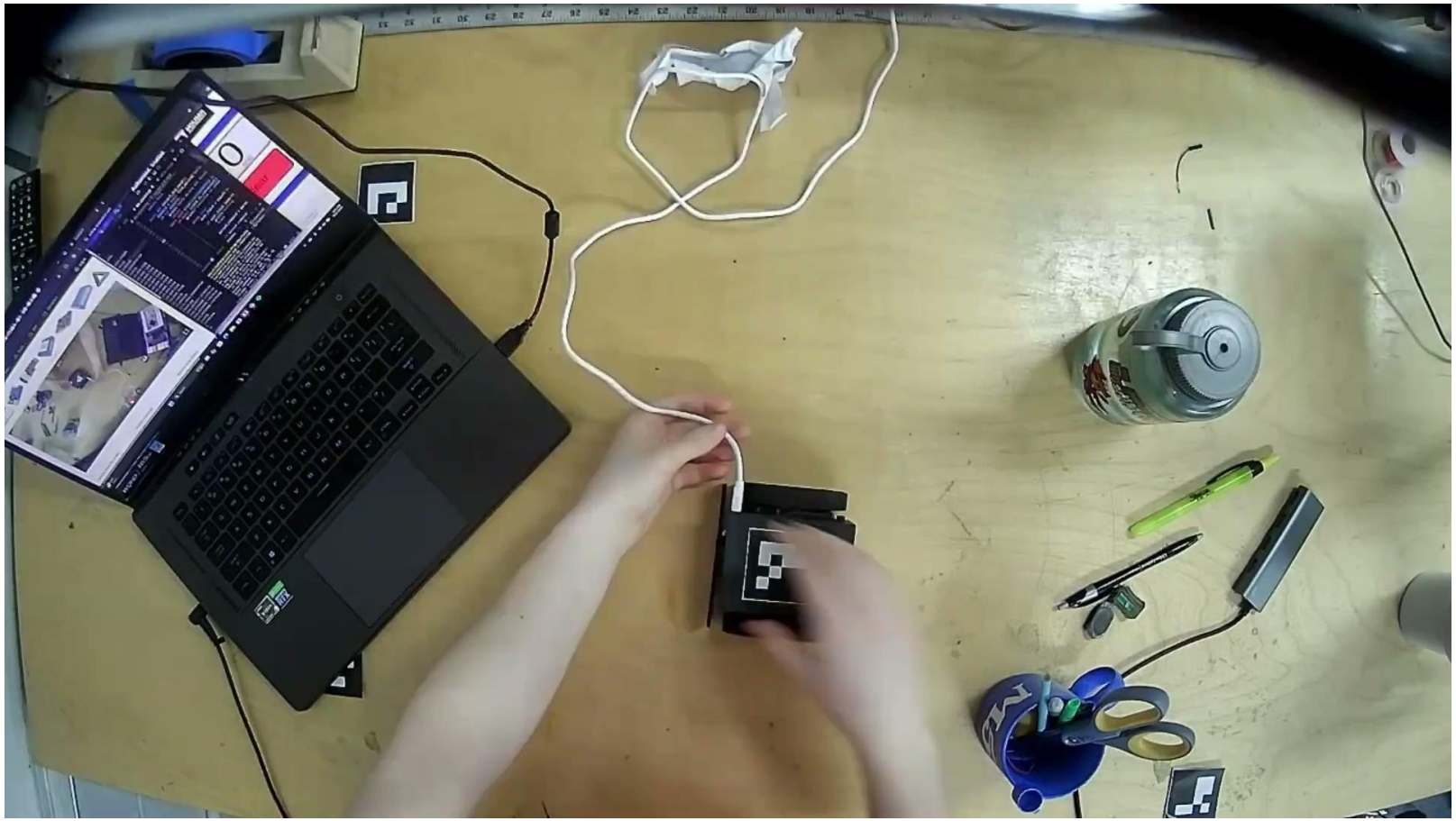
Design Specification 5 - Recognize table boundaries



- The robot identifies 3 fiducial markers (2 corners + robot) and does not start the path planning if fiducials are not identified.
- The robot must be present within the area between the two corner makers.

* Please click on video preview to open link (for PDF)

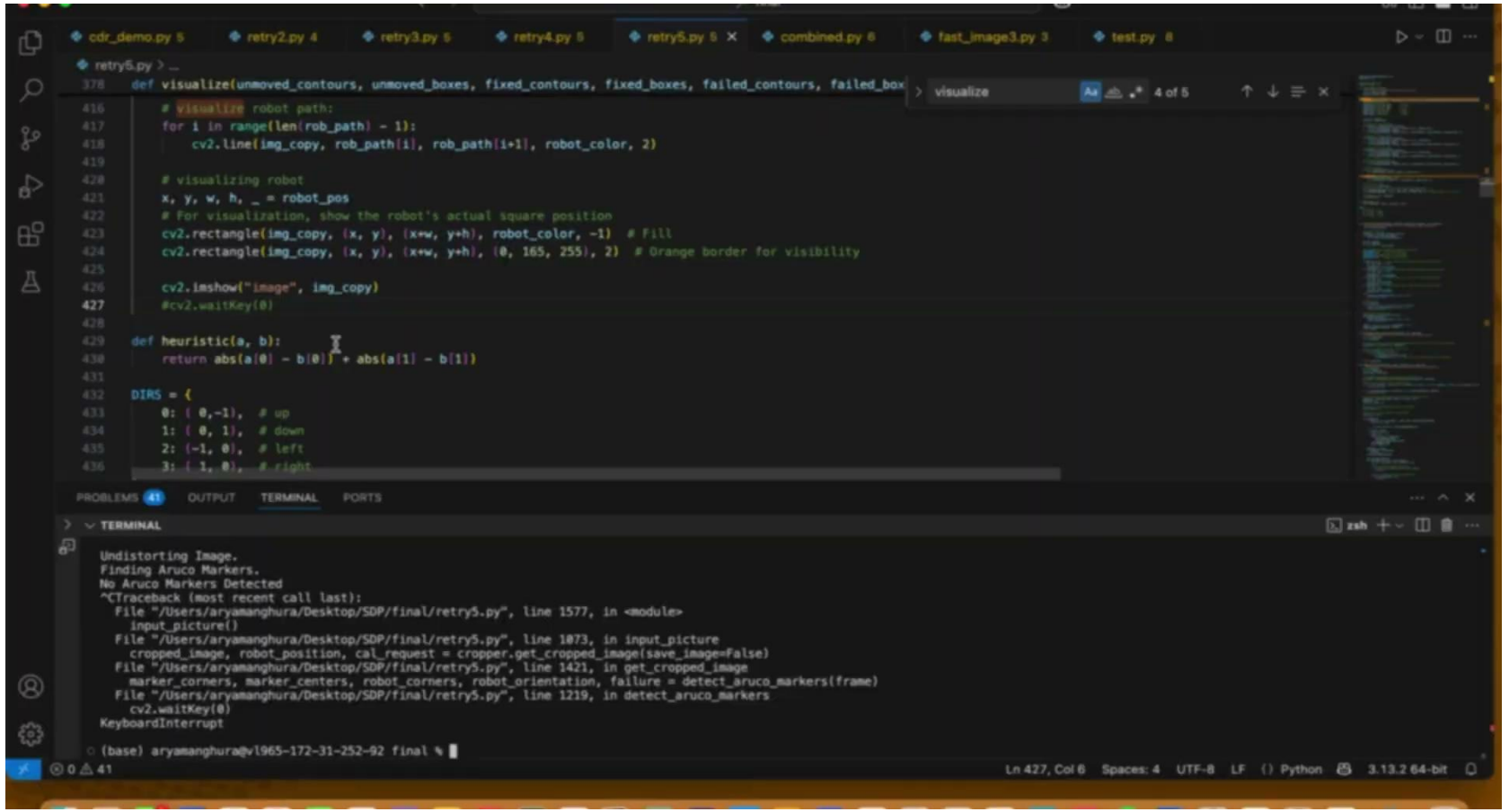
Design Specification 6: 30 Minute Runtime



- The robot is receiving commands to make turns via bluetooth, similar to real world functioning.
- The camera couldn’t capture the initial and final battery %, but it was 100% (4.2V) at the beginning, and 78.33% (~3.94V) at the end
- This gives us a working range of ~1.15 hours.

* Please click on video preview to open link (for PDF)

Design Specification 7- Knoll varying objects



- The robot knolling a tissue box and ignoring the cup (restricted item).
- Top view video of robot in action starts at 00:46 (46 seconds)

* Please click on video preview to open link (for PDF)

Thank you!