Camerazzi

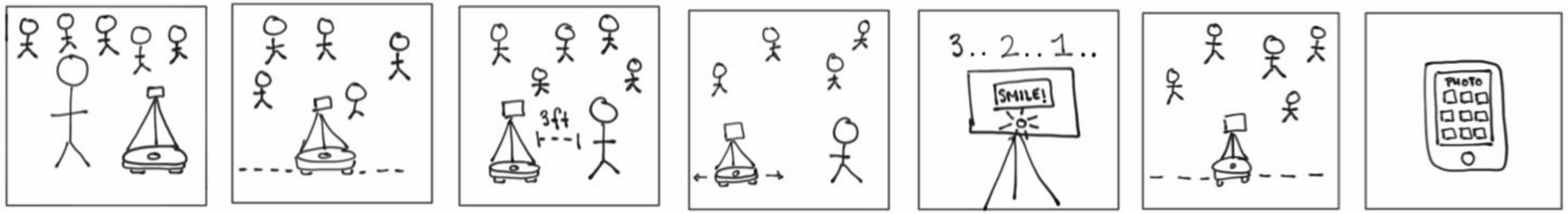
Final Presentation

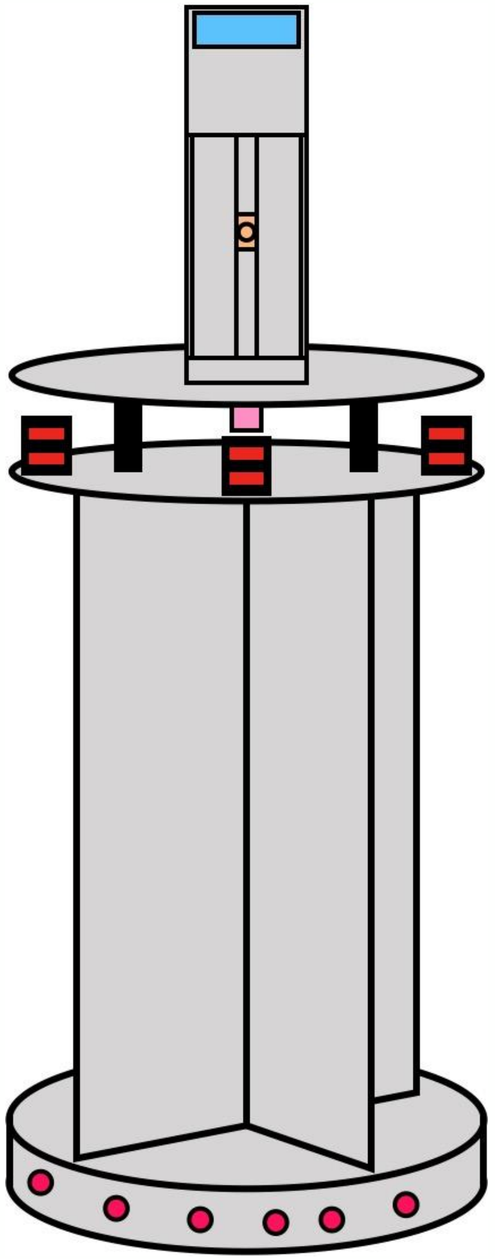
Team B3: Mimi Niou, Cornelia Chow, Adriel Mendoza

Application Area

- Autonomous robotic photographer
  - Comfortable / unintrusive
  - Consistent / unbiased
  - Available
  - Reliable
  - Instant access to photos

- Hardware & Software (& a lot of mechanical engineering)

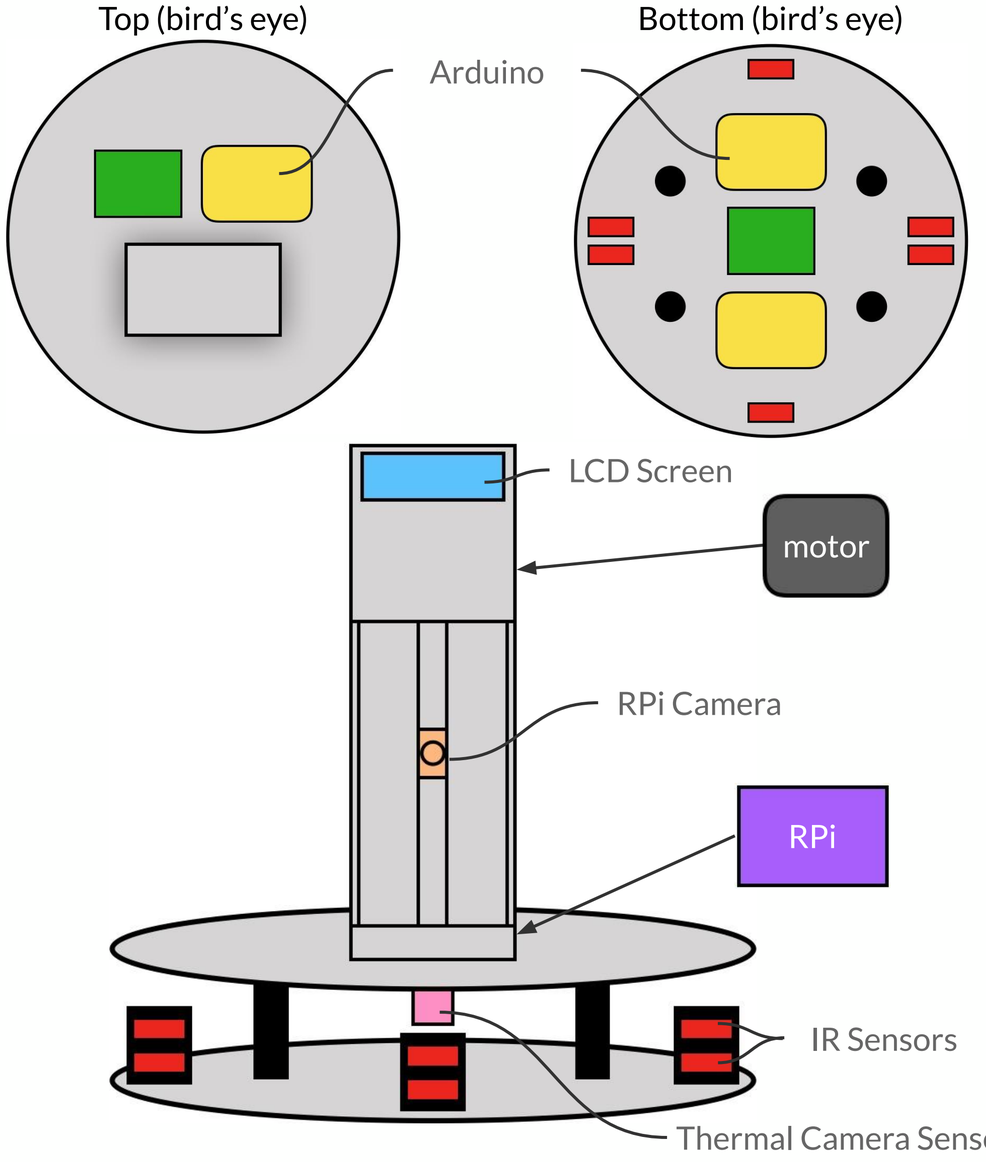




Solution Approach

Hardware:

- iRobot Create 2
- Raspberry Pi 3 Model B
- Raspberry Pi Camera Module V2
- Adafruit AMG8833 8x8 Thermal Camera Sensor
- GP2Y0A02YK0F IR Proximity Sensors (x 12)
- Arduino UNO (x3)
- Motor & motor driver
- LCD Screen

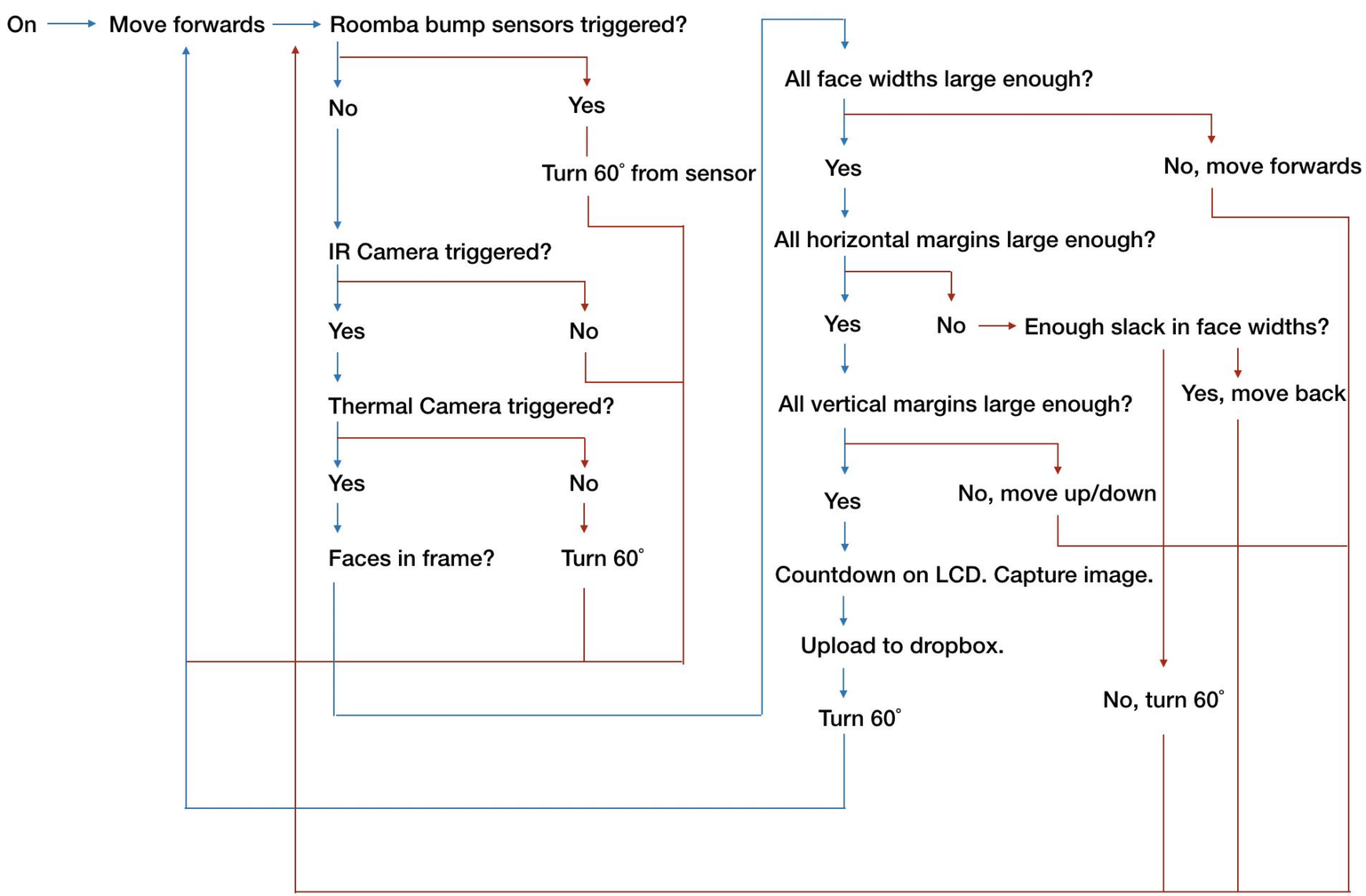


Solution Approach

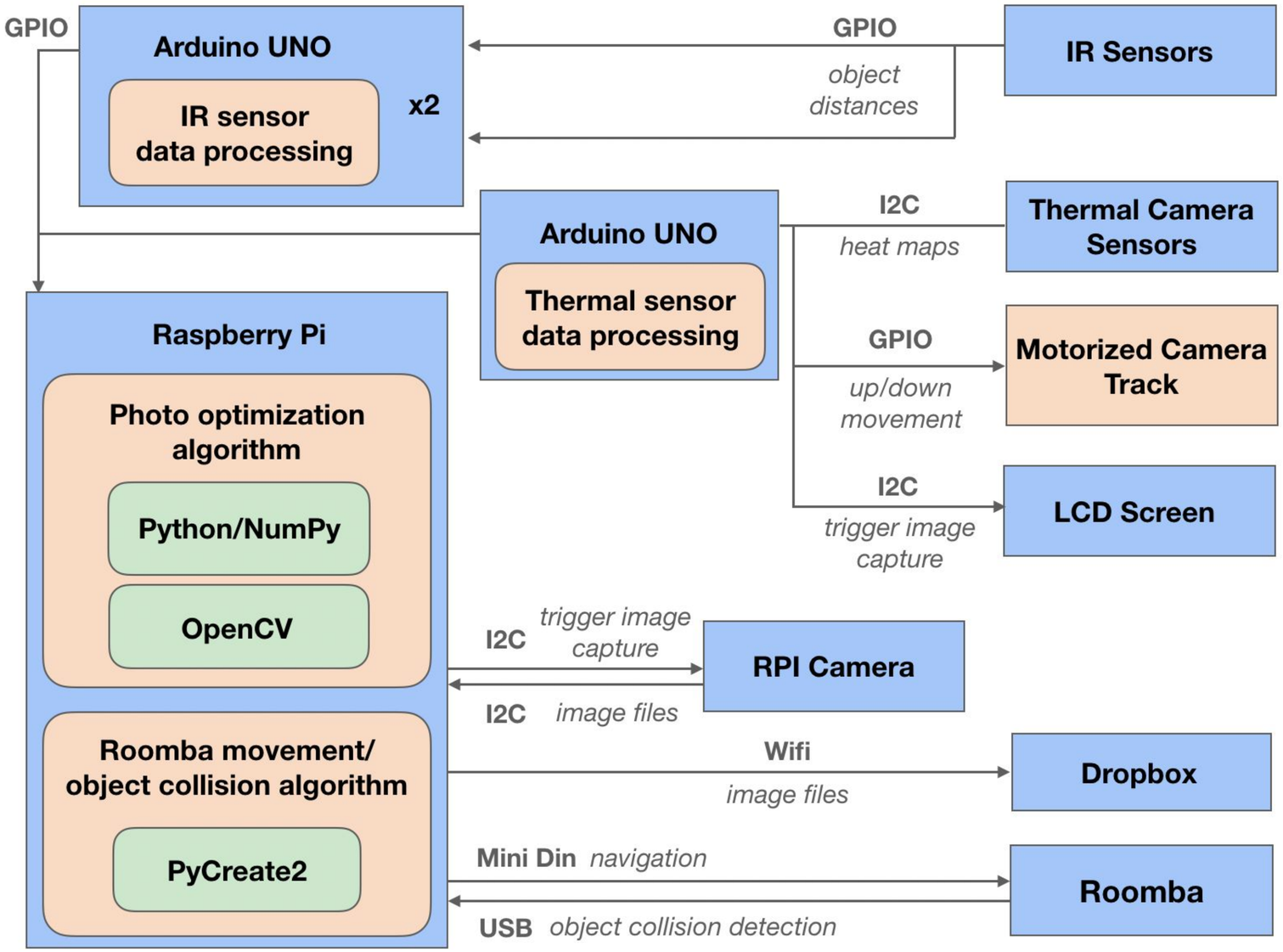
Software:

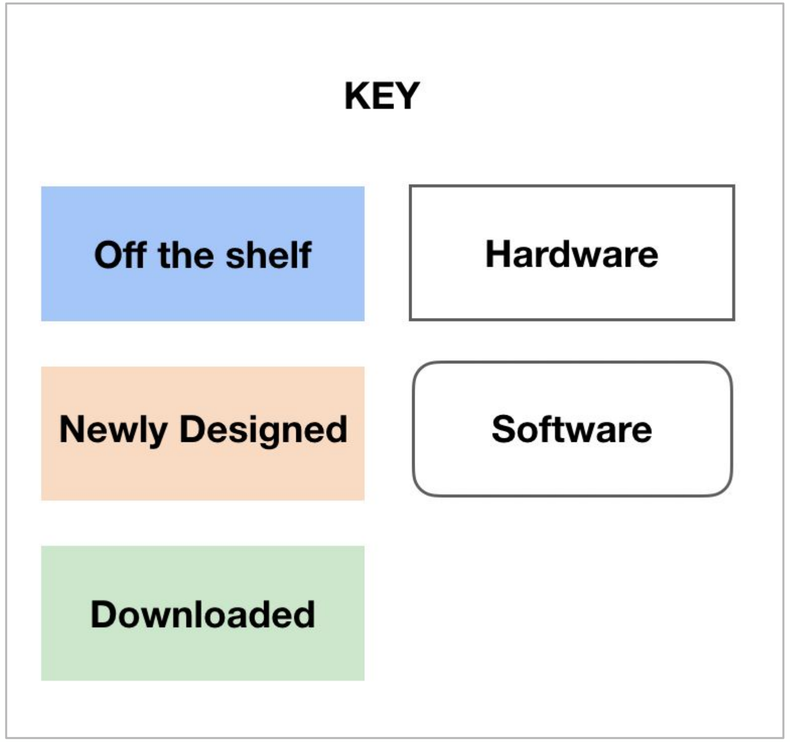
- Python, NumPy, OpenCV for face detection
- Raspbian OS
- Python for thermal camera and IR sensor analysis
- PyCreate2 Library for roomba movement

How it works:



System Specification and Implementation Plan





Metrics and Validation

Tested feature	Metric	Success Value	Tested Value
Face detection	Percentage of faces detected correctly in real time	90%+	83.60%
Photo capture	Percentage of photos with faces	100%	92.70%
Image margins	Moves to optimal position to ensure image margins	5%+ margin all the way around	5%+ margin all the way around
Collision detection	Distance from human when it's detected	At least 3 ft away	18 in away
Roomba stopping latency	Time between human detection and Roomba halting	< 1 sec	< 1 sec
Image transfer	Images wirelessly transferred to designated folders	100%	100%

Metrics and Validation - Face Detection/Photo Capture

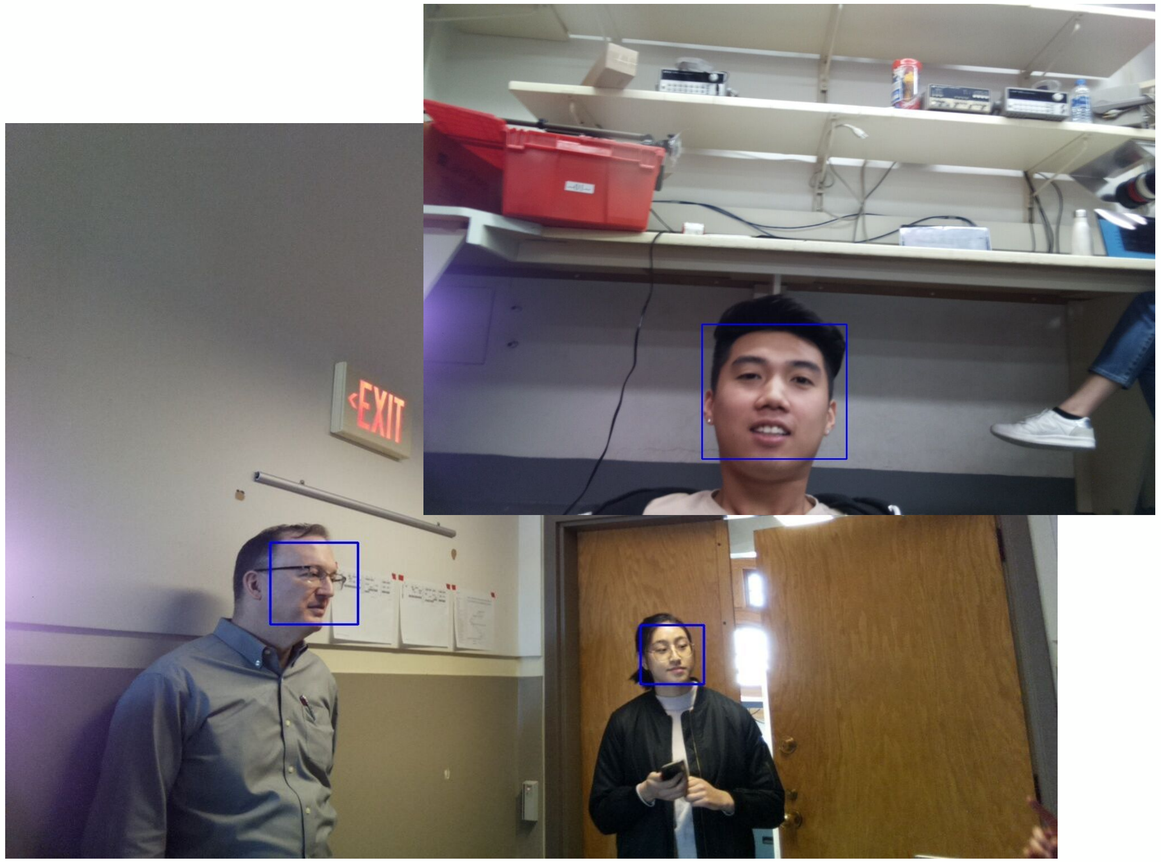
- Testing Approach
  - Let Camerazzi roam in testing environment until 100+ photos taken.
  - Take photos when thermal/IR sensors go off and face detected.
  - Draw a rectangle around all faces identified by OpenCV.
  - Post-analyze photos taken to get testing percentages.

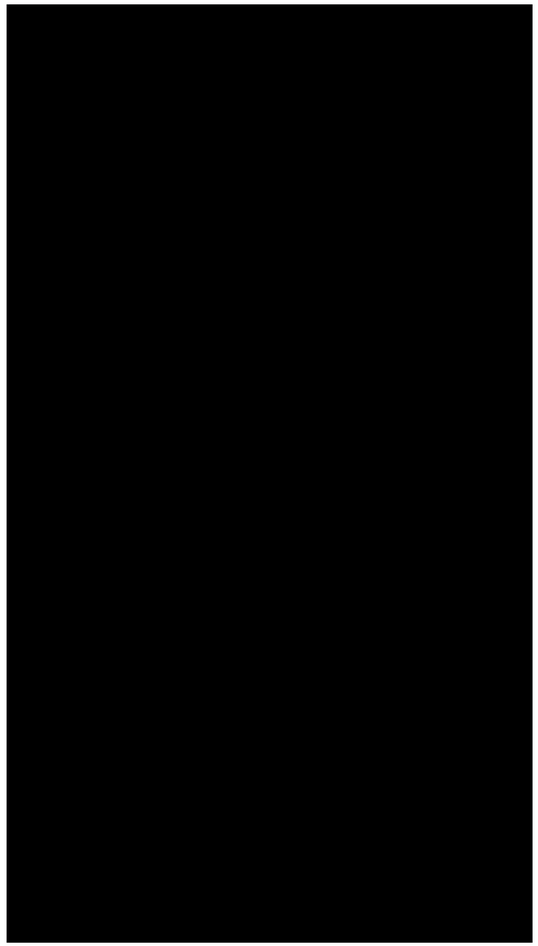
- Face Detection Value: 83.60% - tests OpenCV accuracy
  - # of rectangles outlining faces vs # of rectangles not around faces

- Photo Capture Value: 92.70% - tests overall robot accuracy, with sensors, movement, etc.
  - # of photos which have faces vs # photos without

Metrics and Validation - Image Margins

- Image Margin Success Value: 83.60% - tests image optimization algorithm
  - # of rectangles with enough margin vs # of rectangles without enough margin
  - Analyze uploaded photos' pixel-count around blue boxes





Metrics and Validation - Roomba Movement

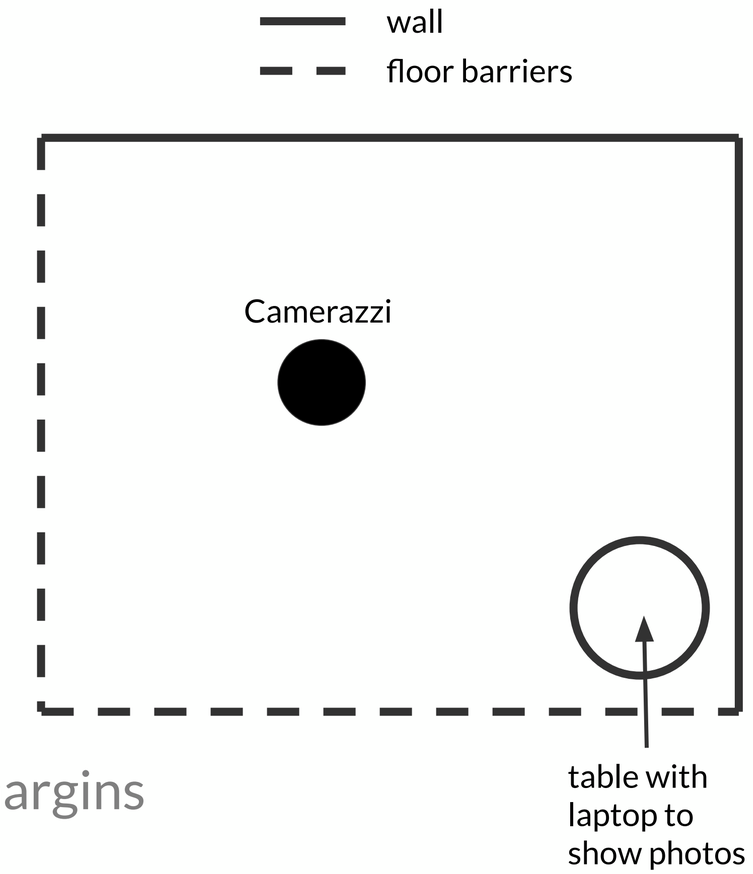
- Collision Detection
  - Testing Approach
    - Set Camerazzi 6ft from a human. Run program which stops roomba when IR and thermal sensors detect human. Measure distance from human torso to roomba. Run from different angles in different environments.
    - Average value over 10 trials: ~18 in.

- Stopping Latency
  - Testing Approach
    - Use stopwatch to time when program prints that human is detected, and when robot stops moving.
    - Average value over 10 trials: ~0.14 s.

Complete Solution

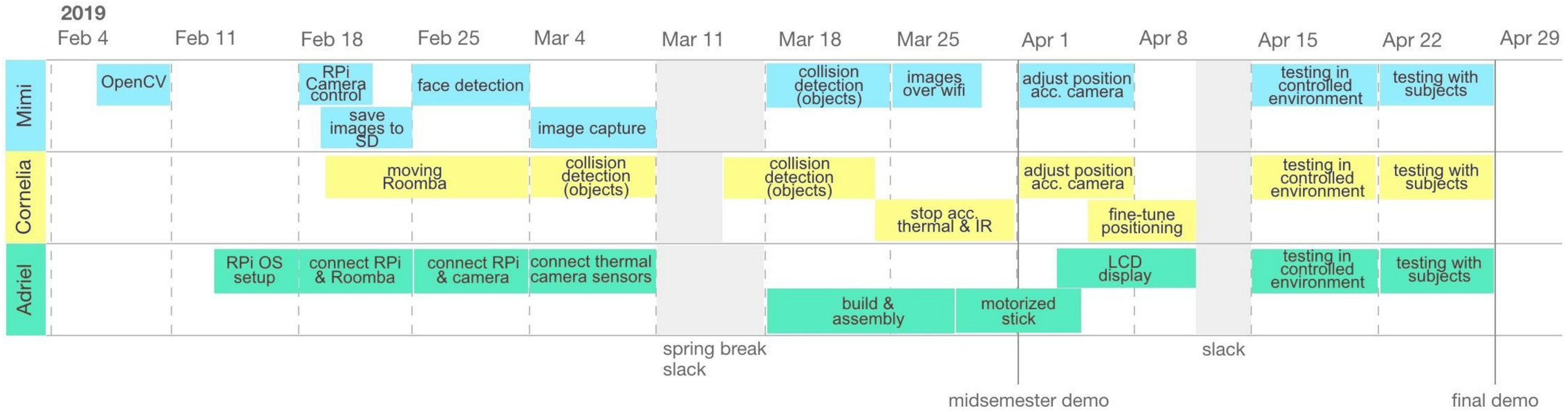
For the public demonstration, our robot will:

- Detect collisions
  - stop at least 1 cm away from walls
  - turn 60° from triggered sensor and continue driving
- Detect humans
  - stop at least 18 inches away from human
- Capture and upload photos
  - detects faces
  - checks margins
  - move camera up and/or move robot backwards to ensure margins
  - capture photo
  - upload to Dropbox folder
  - turn 60° and continue driving



Work Distribution & Schedule

Mimi	Software • Raspberry Pi	• Face detection • Image capture & cloud upload • Photo optimization algorithm
Cornelia	Software/Embedded • Raspberry Pi & Arduino	• Roomba movement • Processing sensor and camera data • Photo optimization algorithm
Adriel	Hardware/Embedded • Arduinos, sensors, & structure	• Connecting & powering components • Robot mechanics & assembly • Sending sensor data



Remaining Work & Lessons Learned

Remaining work before public demo:

- Eliminate risk of loose wires by soldering connections that are currently through breadboard
- Test robot with lighting conditions & WiFi in Wiegand Gym
- Create floor barriers for public demo

Lessons learned:

- Try to scavenge the parts you are looking for from other projects, classes, departments
- Have a clear understanding of how long each specific task will take
- Don’t procrastinate on filling out order forms
- Test out sensors before ordering them in bulk to ensure that they meet your design requirements