Project P.E.T.E

Procedure Execution Tracking Engine

Spencer Tang, Sophie Guan, Frank Yao, Anoushka Sawant, Aaron Sin

Table of contents

01 Problem Overview

- Summary of problem
- Potential applications

02 Solution Overview

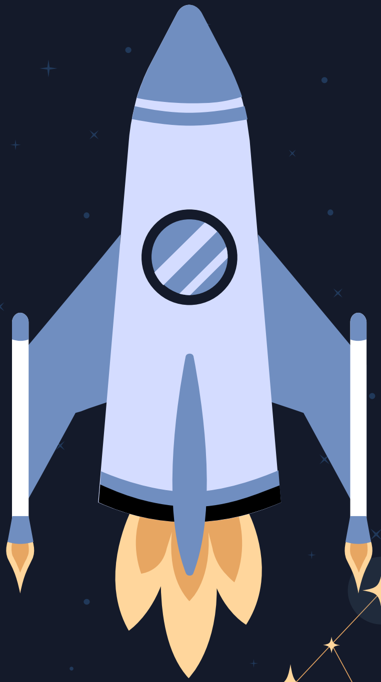
- Summary of solution
- Block diagram
- Component list

03 Project Logistics

- Responsibilities
- Timeline
- Current progress

04 Risk Analysis

- Critical components
- Possible future issues



Problem Summary

- **Goal**: Monitor a user's progress as they complete a procedure

- **Requirements**:
    - tracking completion of each step in the procedure
    - evaluating correctness
    - identifying deviations

- **Procedure:** removing the bottom bracket of a bike.

Application & Significance

NASA: Replacing ground assistance with computer assistance

Medical: Assisting with surgeries

Education: Help with learning & repeatedly practicing a procedure

General Significance:

- Safety & Accuracy (best practice, identifying deviations)
- Efficiency (computerized assistance)

Procedure

Why a bike?

- Distinct features that could be recognized
- Variety with t

What is a bott

The bicycle co

kset

Procedure

Removing Bike Crank

1. Remove bike pedal (Optional)
2. Remove bike chain (Optional)
3. Identify crank standard

Self-Extracting

2-piece

3-piece

Procedure

Removing Bike Crank

1. Remove bike pedal (Optional)
2. Remove bike chain (Optional)
3. Identify crank standard
4. Identify required bike tools

Crank Remover

Socket Wrench

Monkey Wrench

Procedure

Removing Bottom Bracket

1. Identify Bottom bracket

Threaded Shell

Press Fit Shell

Thread-through Shell

Procedure

Removing Bottom Bracket

1. Identify bottom bracket
2. Identify threading method

External Notches


Pin-holes


Internal Notches

Bracket

m bracket

ding meth

3. Identify bracket tool

External Notches

Pin-holes

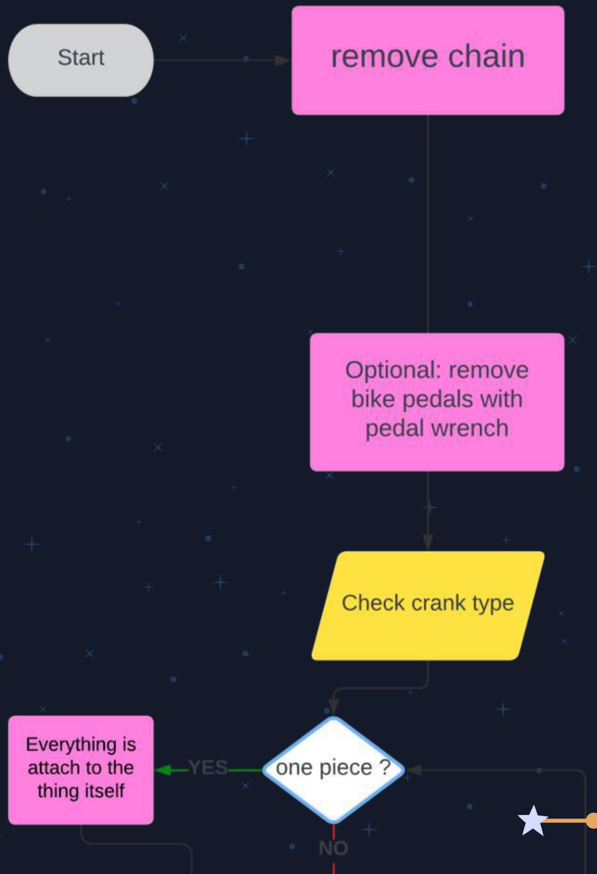
Internal Notches

Solution Summary + Critical Parts

The Big Idea

1. Given a predefined step list
2. User performs each step
3. Validate (using computer vision and sensor data) if the step is correctly done

Steps:



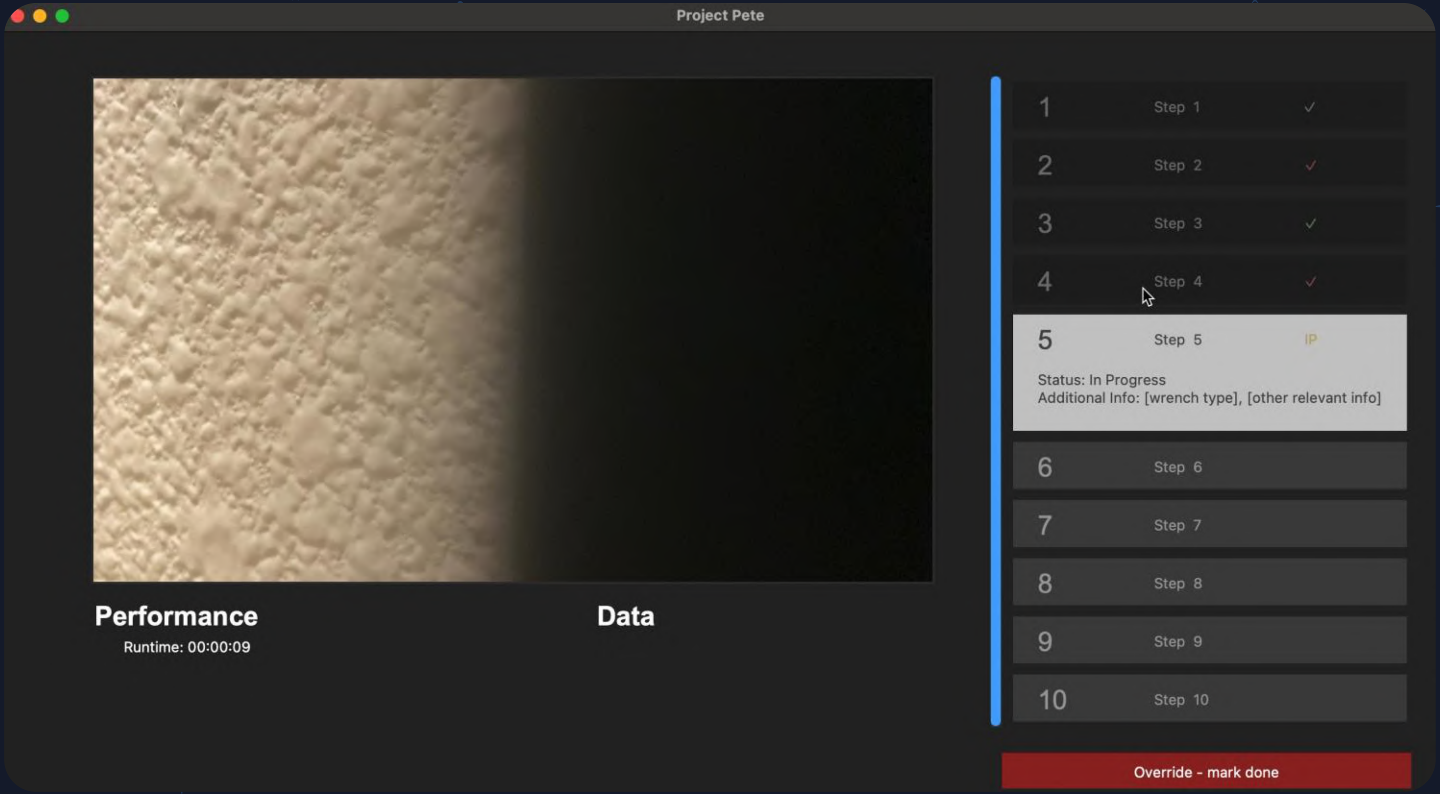
Criteria for 'Correctly Done':

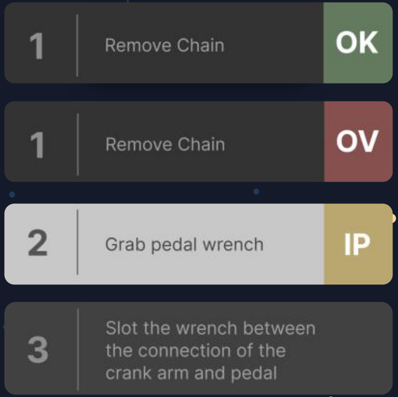
Identify chain and presence & orientation on bike,

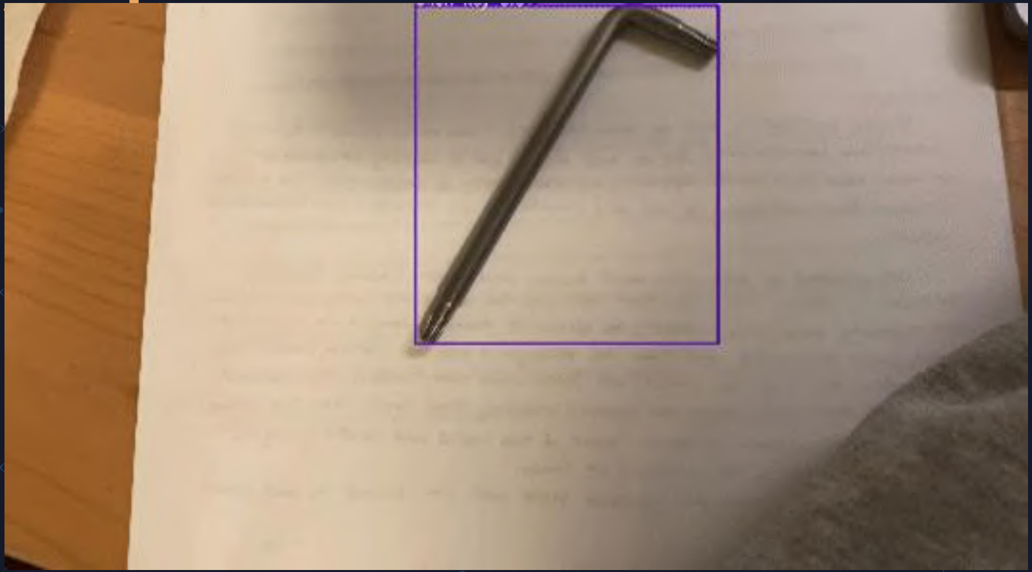
identify bike pedals, large chain ring (for safety), crank arm, and pedal wrench, identify left or right pedal, check for proper loosening direction (right pedal loosens counterclockwise, left pedal loosens clockwise), choose position with best mechanical advantage

Identify crank: check for lightening shape (one), check for left crank arm and the gear+right crank arm+crank spindle (two), Left crank arm, right crank arm, center crank spindle (three)

(user interface demo)







(very, very preliminary model demo)

Validation Data:

1) Computer Vision

Purpose: identify objects of interest
(Ex: hand, Allen key)

2) Sensors

Purpose: supplement CV with details (e.g. rotation, temperature, torque)
(Ex: Allen key rotation)

1) Computer Vision (Model)

Task: Real-Time Object Recognition

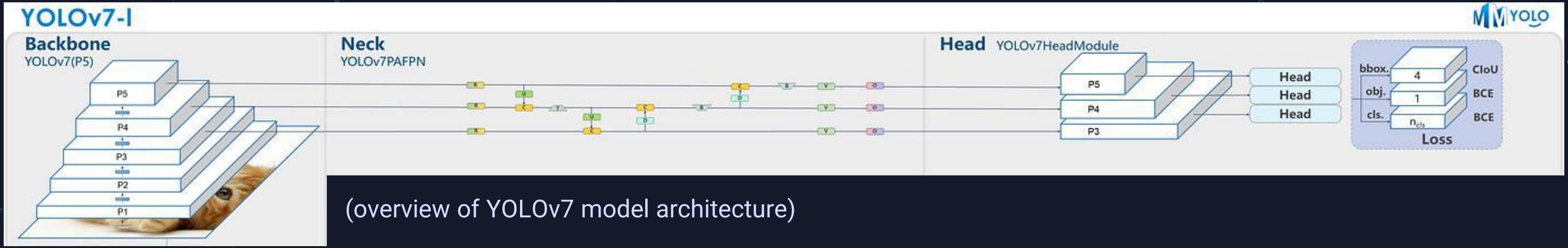
Model: YOLOv7-Tiny (You Only Look Once)

→ **YOLOv7-Tiny** model (built for Edge Devices) with 6M parameters

vs. **YOLOv7** model with 37M parameters

→ Faster computations, less resource-intensive, higher FPS (17 vs. ~6)

→ But lower accuracy

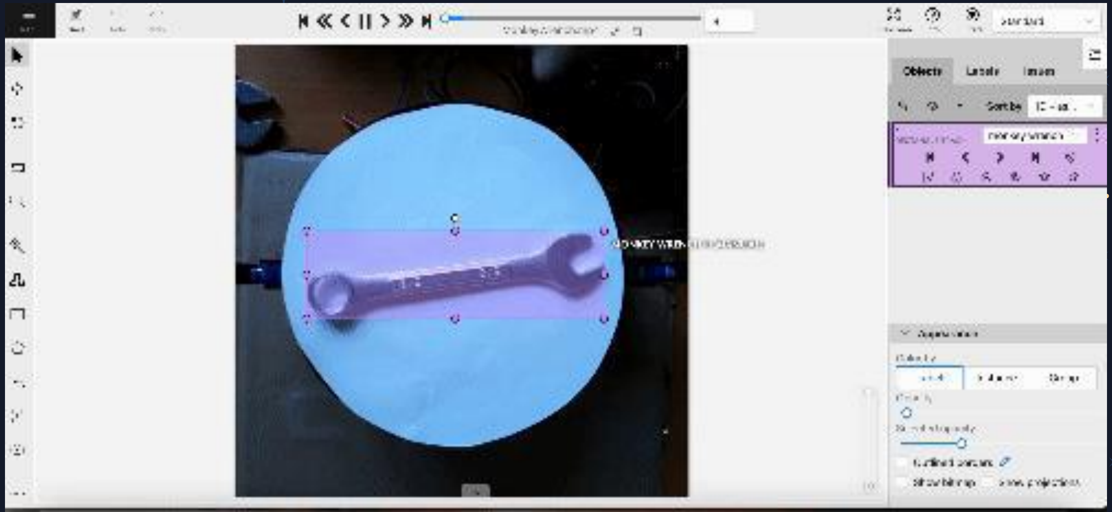


(overview of YOLOv7 model architecture)

1) Computer Vision (Data)

Current dataset classes:
- Hands
- 7 different classes of bike tools
- **Caveat:** not enough images per class for a robust model

Solution:
- Image Augmentation
- Few-Shot Learning via fine-tuning



(an example labeling of a monkey wrench on CVAT)

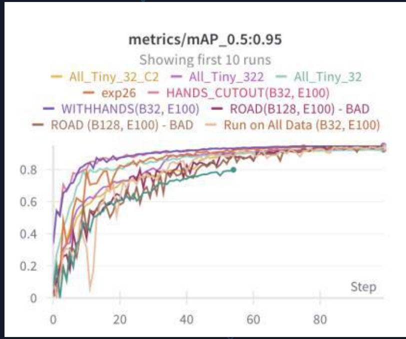
1) Computer Vision (Training)

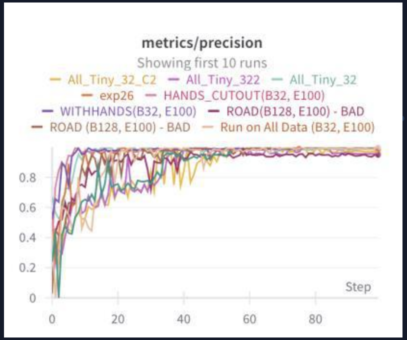
Training: POD clusters GPU nodes (& large memory nodes for >64 batch size)

CSC Center for Scientific Computing

System Hardware		
	CPU count	24
	GPU count	4
	GPU type	Tesla V100-SXM2-32GB

Hyper-parameters: batch/epoch, image augmentation %, loss, anchors, learning rate





2) Sensors

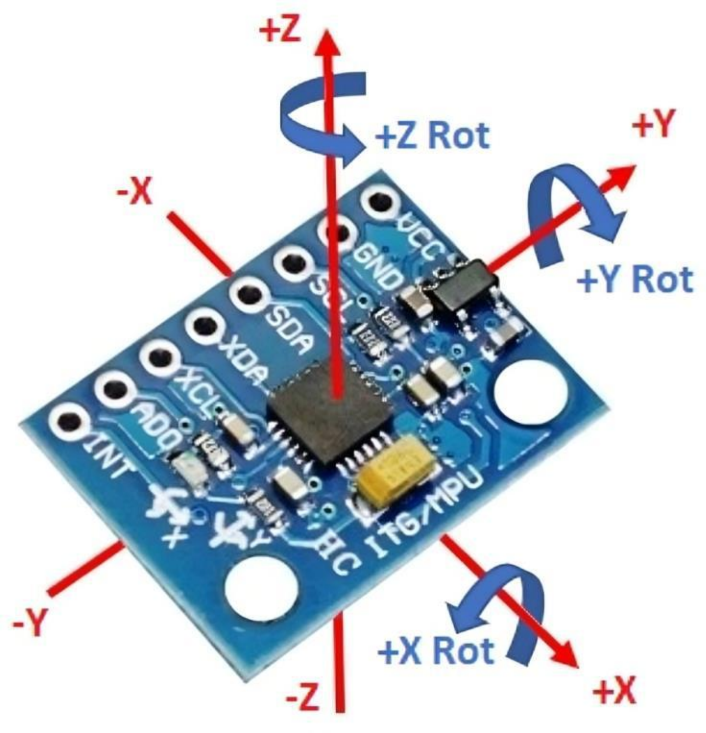
Smart Tools

- Assists the CV Model

MPU-6050 Sensor

- 6-axis Accelerometer/Gyroscope
- GY-521 breakout board

- Accelerometer -> X, Y, Z
- Gyroscope -> Pitch, Yaw, Roll



Validation Logic:

Intersection of Bounding Boxes

- Determine procedure's start and end point

  Overlapping → Not overlapping

  Not overlapping → Overlapping

- Threshold: Percentage of Intersection

```
If  IoU > threshold:
     then: start/end procedure
```

- Shortcomings
  - Correctness cannot be determined → Sensors!!

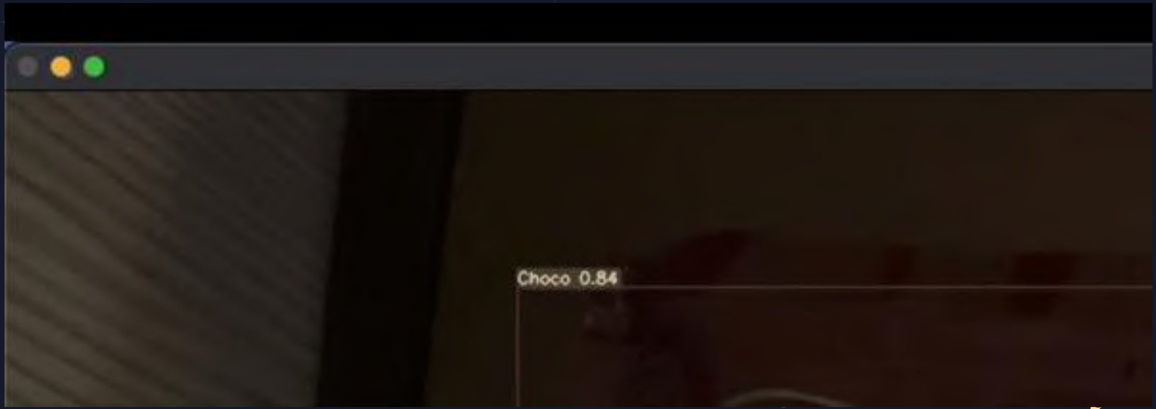
What can be used?

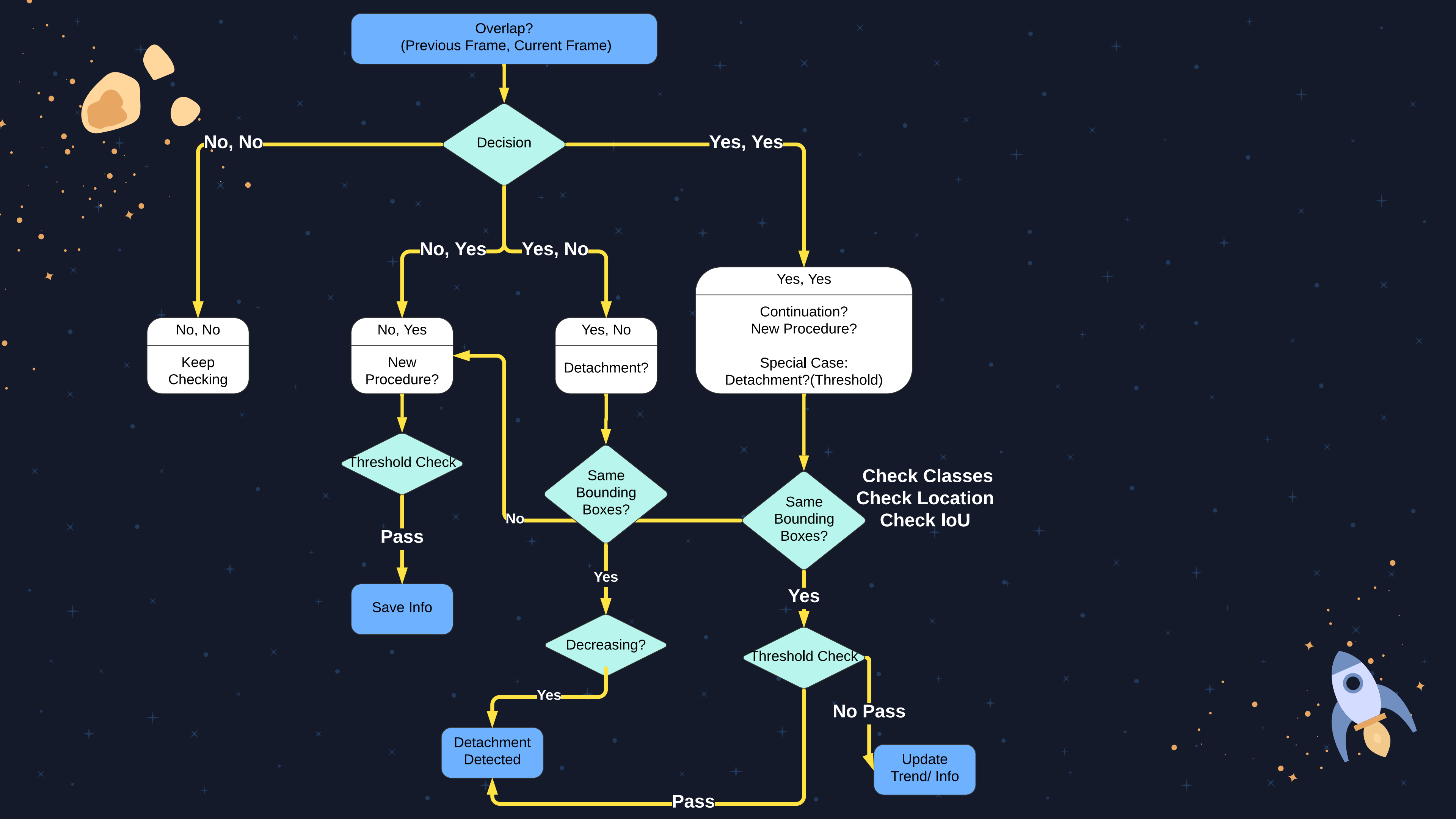
1) Intersection over Union
2) IoU Trend
3) Number of (Overlapping) Boxes

    Limiting Factor

    Attachment/Detachment Detection

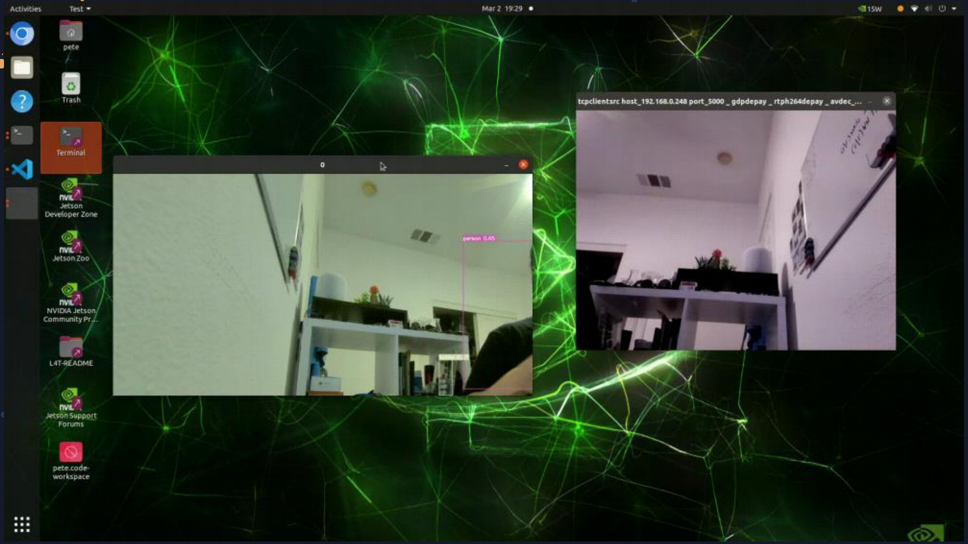
4) Euclidean Distance
5) Detected Class



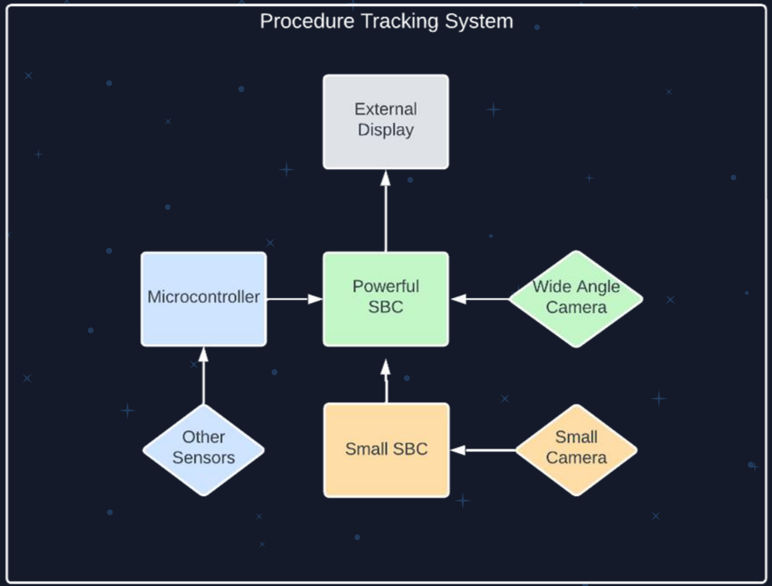

Overlap?
(Previous Frame, Current Frame)
Decision
No, No
Yes, Yes
No, Yes
Yes, No
No, No
Keep Checking
No, Yes
New Procedure?
Yes, No
Detachment?
Yes, Yes
Continuation?
New Procedure?
Special Case:
Detachment?(Threshold)
Threshold Check
Pass
Save Info
Same Bounding Boxes?
No
Yes
Decreasing?
Yes
Detachment Detected
Same Bounding Boxes?
Check Classes
Check Location
Check IoU
Yes
Threshold Check
No Pass
Update Trend/ Info
Pass


meiji
CHOCOROOMS
Sweet & Crispy

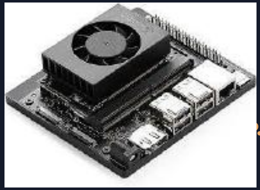
Procedure Tracking System Hardware Demo



Block Diagram



Block Diagram

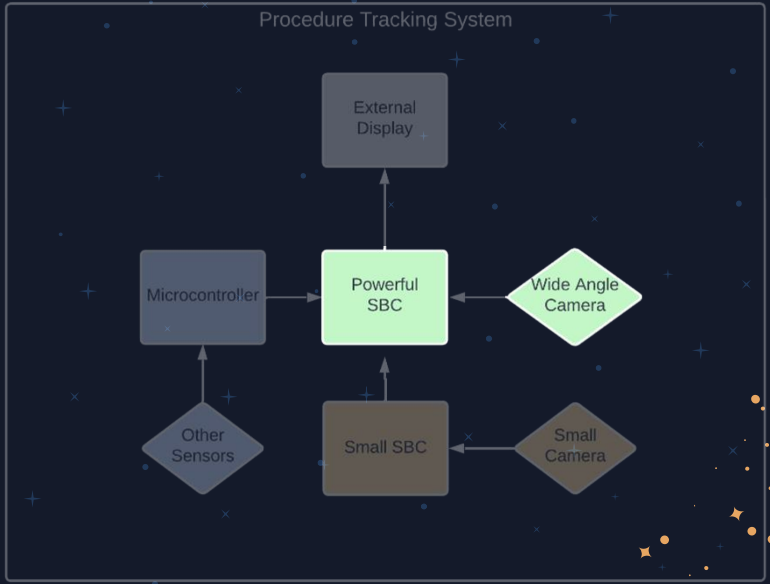


NVIDIA Jetson Orin Nano

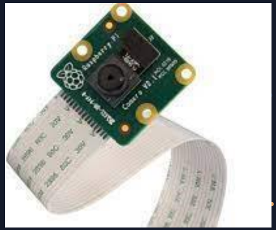
Powerful Single Board Computer (SBC) with the processing power to run our image recognition model

E-con81 Camera

High fidelity, wide angle camera used to monitor the overall view of the workstation



Block Diagram

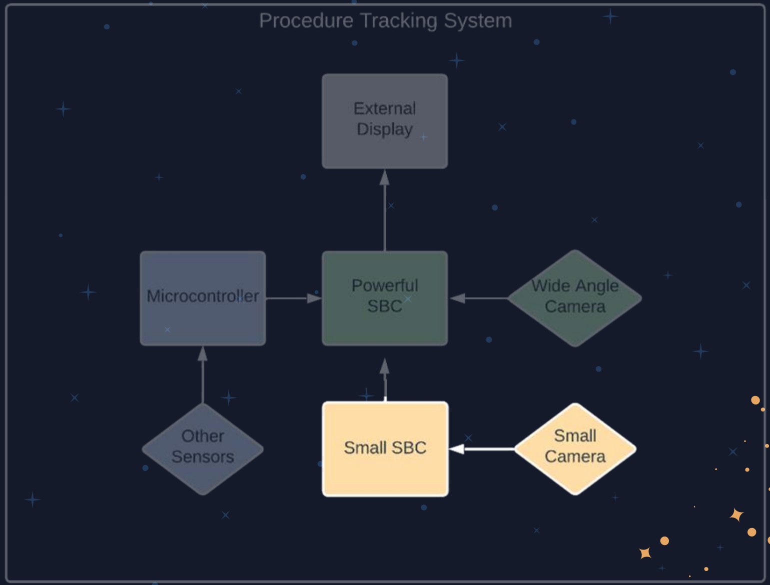


Pi Camera Module 2

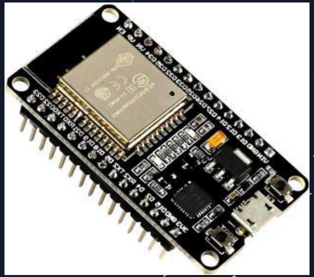
Used to get a user-mounted camera angle

Pi Zero W

Due to its lower power consumption and wireless capabilities, we can wirelessly stream the video to the SBC running our model



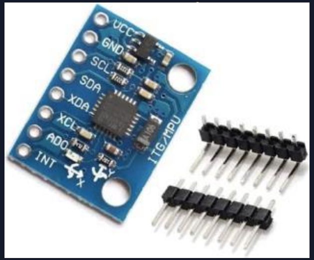
Block Diagram



ESP 32s

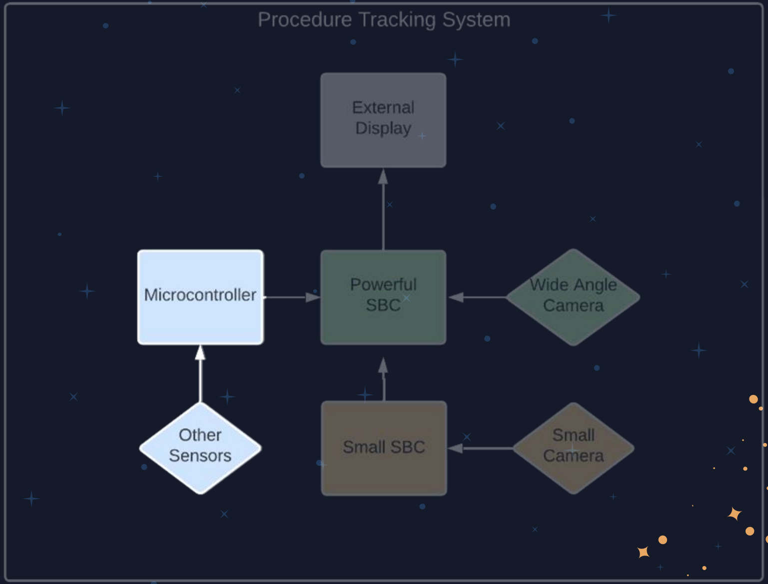
Microcontroller used to interface with additional sensors for procedure tracking

Equipped with Bluetooth and wireless capabilities to allow for wireless sensor usage



GY-521 MPU-6050

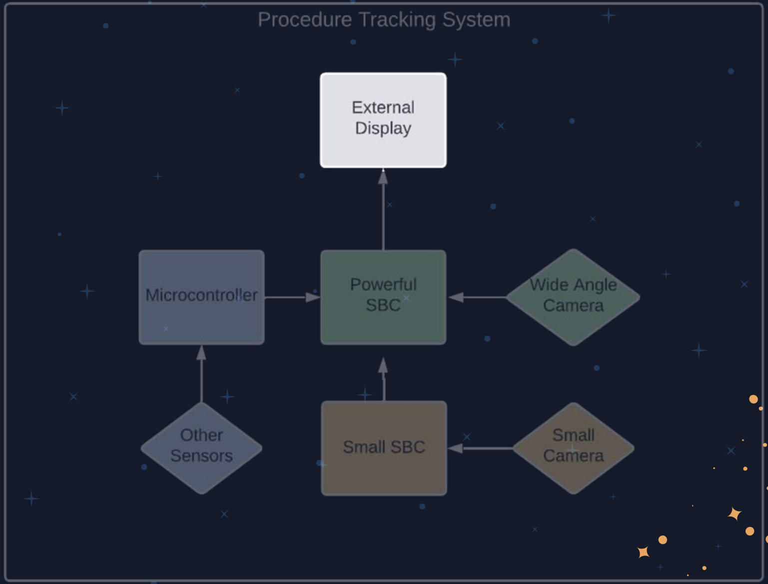
6-axis accelerometer gyroscope sensor module used to track movement of tools where camera data is not enough



Block Diagram

External Display (Monitor)

Used to notify user of current progress of the tracked procedure through a GUI to-do list



Team Member Responsibilities

Spencer

- Sensor Interfacing
- IoT Tools
- Data collection

Sophie

- CV model training + improvement
- GUI

Frank

- SBC and Camera interfacing
- Data streaming and detection pipeline

Anoushka

-Logic Design & Integration with GUI
-Data Labeling

Aaron

- CV model training
- Logic/Decision & Integration with GUI

Current Progress + Plans

WINTER

Label All Data

Continue to collect more data from rig and online and label it

Train and Test Mdel

Train and test model for accuracy and performance

Install & Test Sensors

Determine which sensors to use and add them

Deploy Model

Deploy model onto board

SPRING

Integrate Sensors with CV

Create validation logic for each procedure step

Assemble Display

Assemble GUI/display for procedure display

Wrap

Test run and see if everything works

Risk Analysis & Key Challenges

Model accuracy:

- General performance
- Obstruction
- Noise (background)

Validation accuracy:

- Correctly identifying ‘interaction’ from bounding box intersection
- Resolving conflicting information from CV and sensors

Thanks for Listening!

& thank you to our mentors **Jessica Marquez and John Karasinski**
TA **Alex Lai**
Professor **Yoga Isukapalli**

Any Questions?