Final Presentation

Senior Design 491: Soybean Parasitic Cyst Detector

Team: 10

Client: Dr.Pandey, Santosh

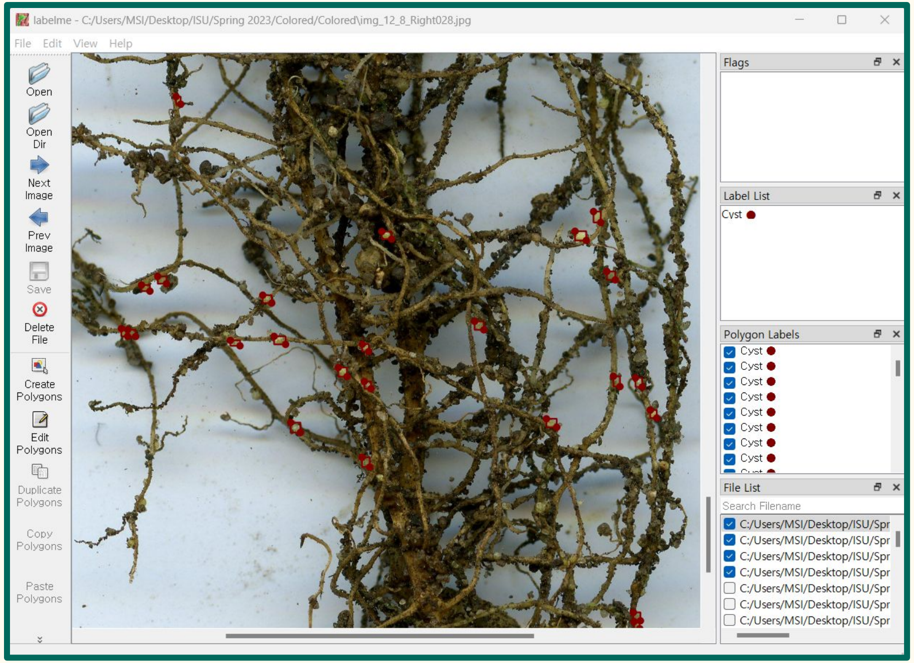
Chris Cannon, Ethan Baranowski, Katherine Moretina, Matthew Kim

Problem Statement

- To determine how many parasitic cysts are on the roots of soybean plants, we will develop a deep learning algorithm designed for small object detection.
- We will also create a device to integrate image capturing with the machine learning algorithm.
- Goals:
  - Increases productivity in farms.
  - Allows geneticists to breed resistant soybeans.

Using Machine Learning to Detect Cysts

- Machine learning is commonly used to detect objects in images
  - Self-driving cars
  - Drone footage
- Using an algorithm known as Faster Recursive Convolutional Neural Network.
  - (Referred to as Faster R-CNN)
  - 2-Stage Algorithm



Mask R-CNN: Optimized Faster R-CNN

- Compared to other algorithms, Mask R-CNN:
  - Has comparable or better accuracy
  - Has higher accuracy for small object detection (better at identifying birds in a flock)
  - Runs moderately slower
- Two Neural Networks in One:
  - Region Proposal Network
  - Classifier Network
- Two-stage Algorithm:
  - Increased accuracy over one-stage.
  - Second Stage adds RoI (Region of Interest) Pooling layer to filter results.

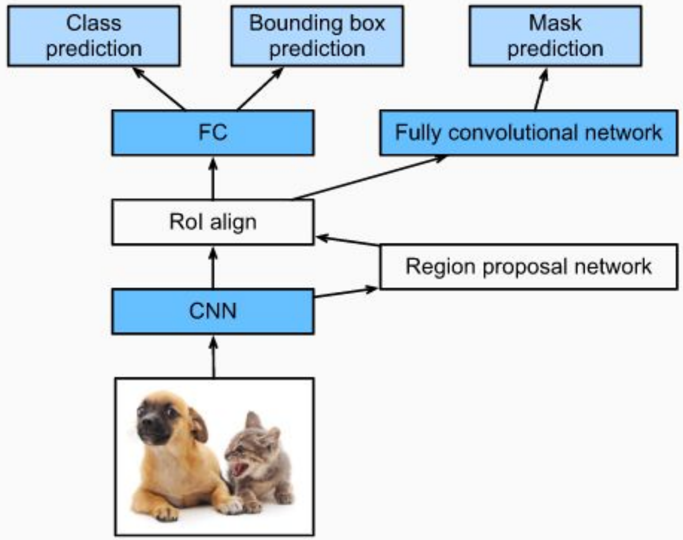
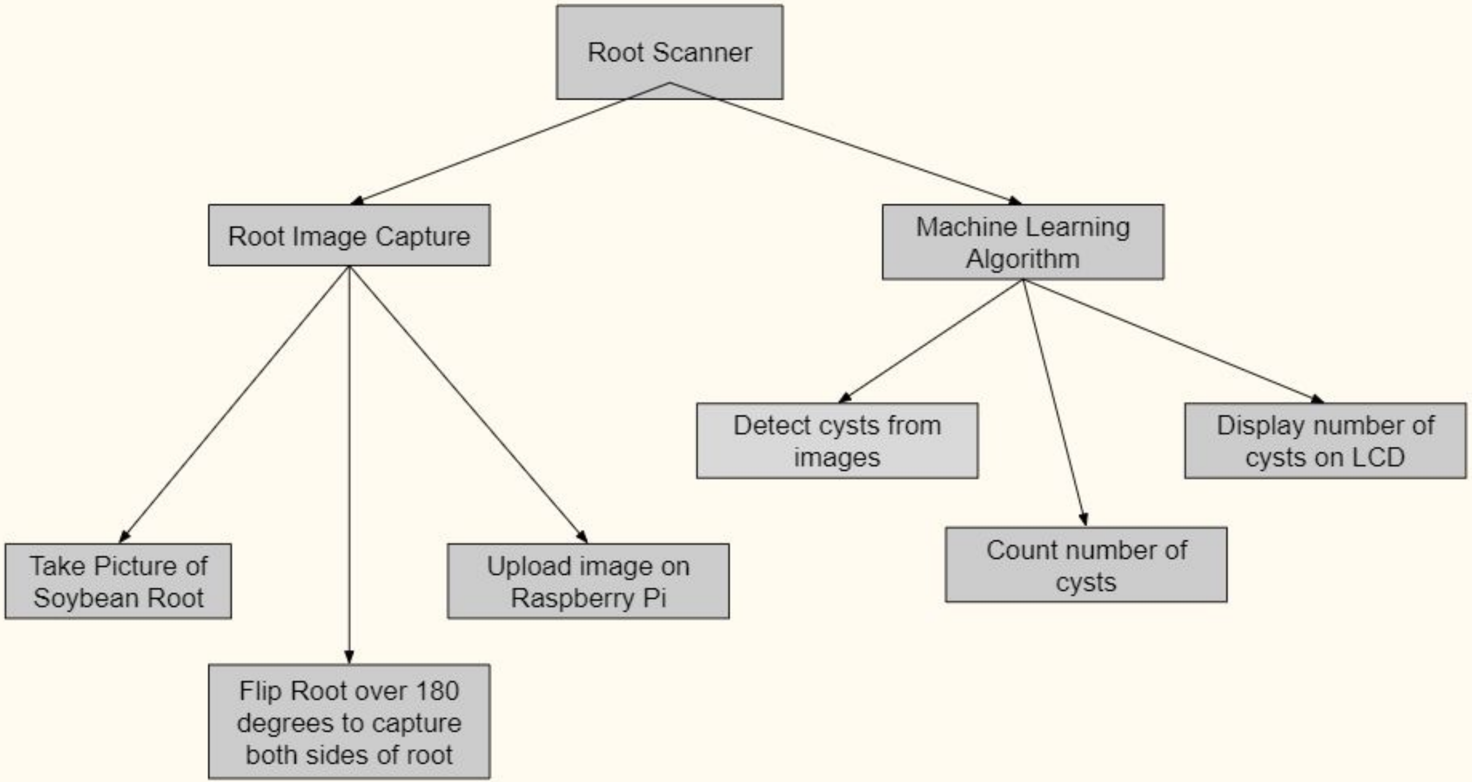


Fig. 14.8.5 The mask R-CNN model.

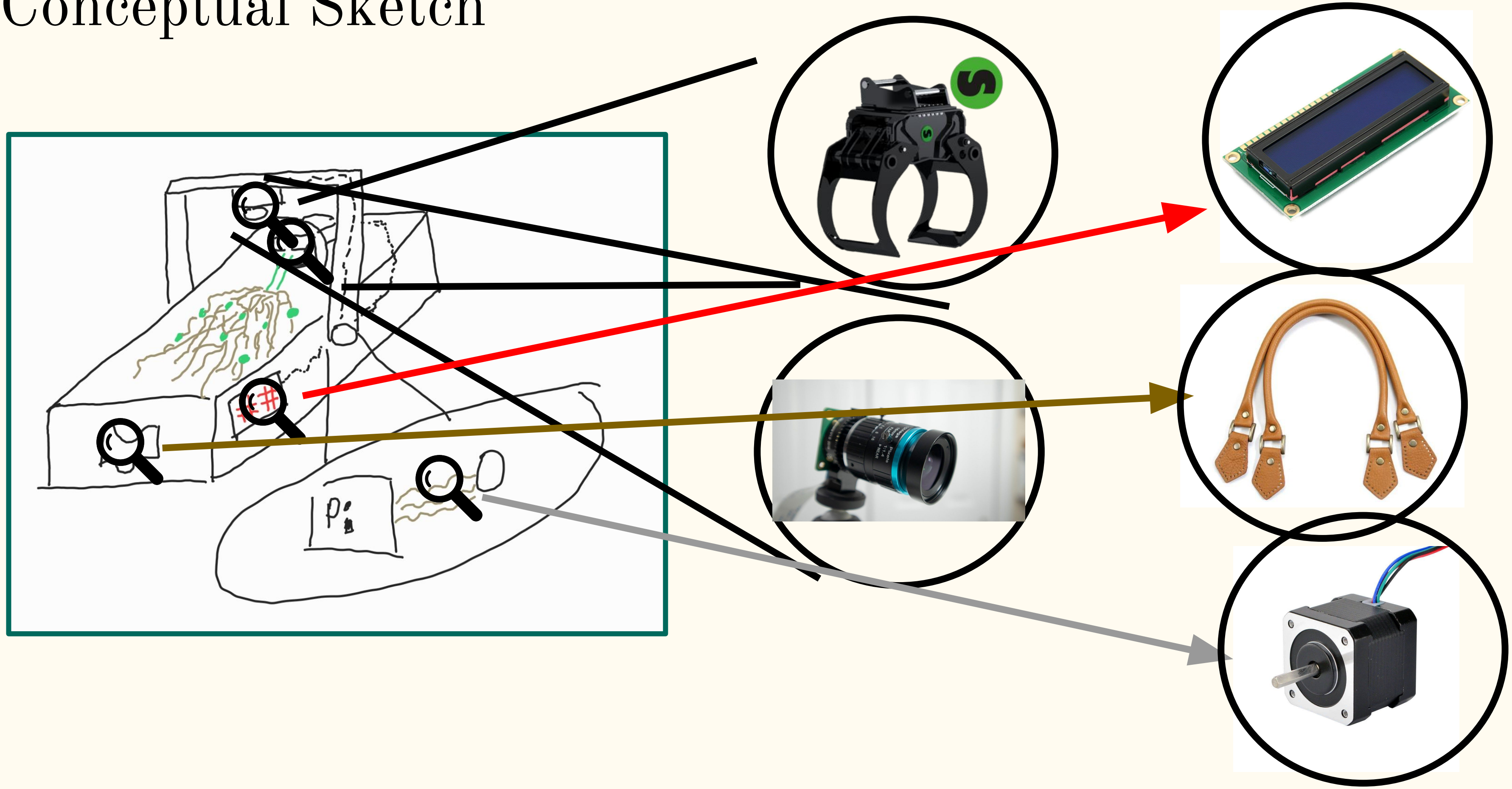
For more information how we chose our object detection algorithm, see Appendix slide "Faster R-CNN vs. YOLO vs. SSD"

Functional Decomposition

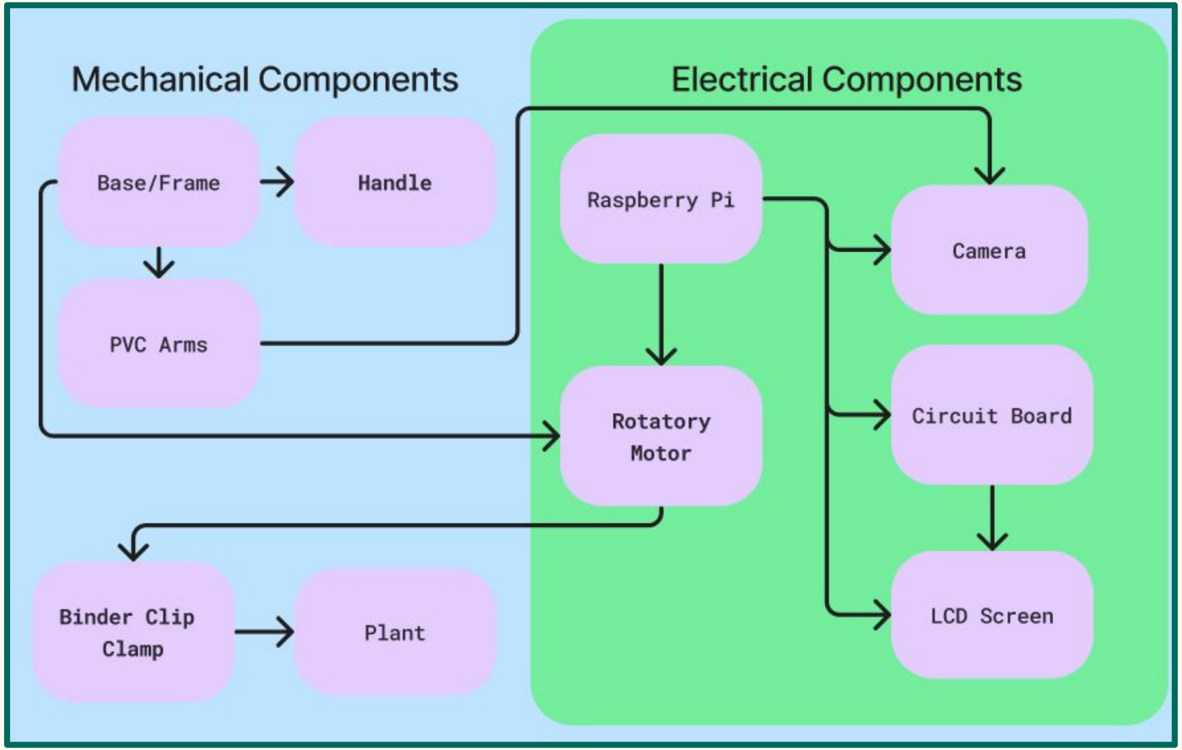


https://steelwrist.com/ko/products/grapples/
https://www.youtube.com/watch?v=_Oh7sITnwMU

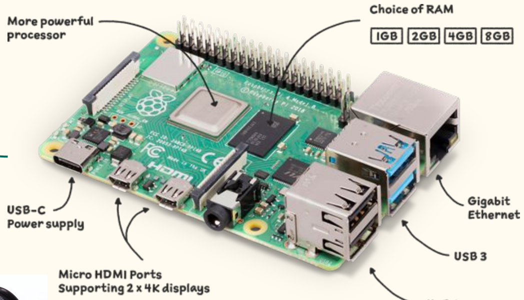
Conceptual Sketch

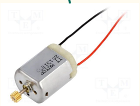


Detailed Design



Hardware, Software, And Technology Platforms Used





Hardware

- Raspberry Pi 4
- Camera
- LCD Screen
- Motor

Software

- Detectron2's Mask-RCNN implementation
  - Provided by Facebook's research team under the Apache 2.0 License

Platforms

- Google CoLab
  - Provides computing resources to train a Detectron2 model.

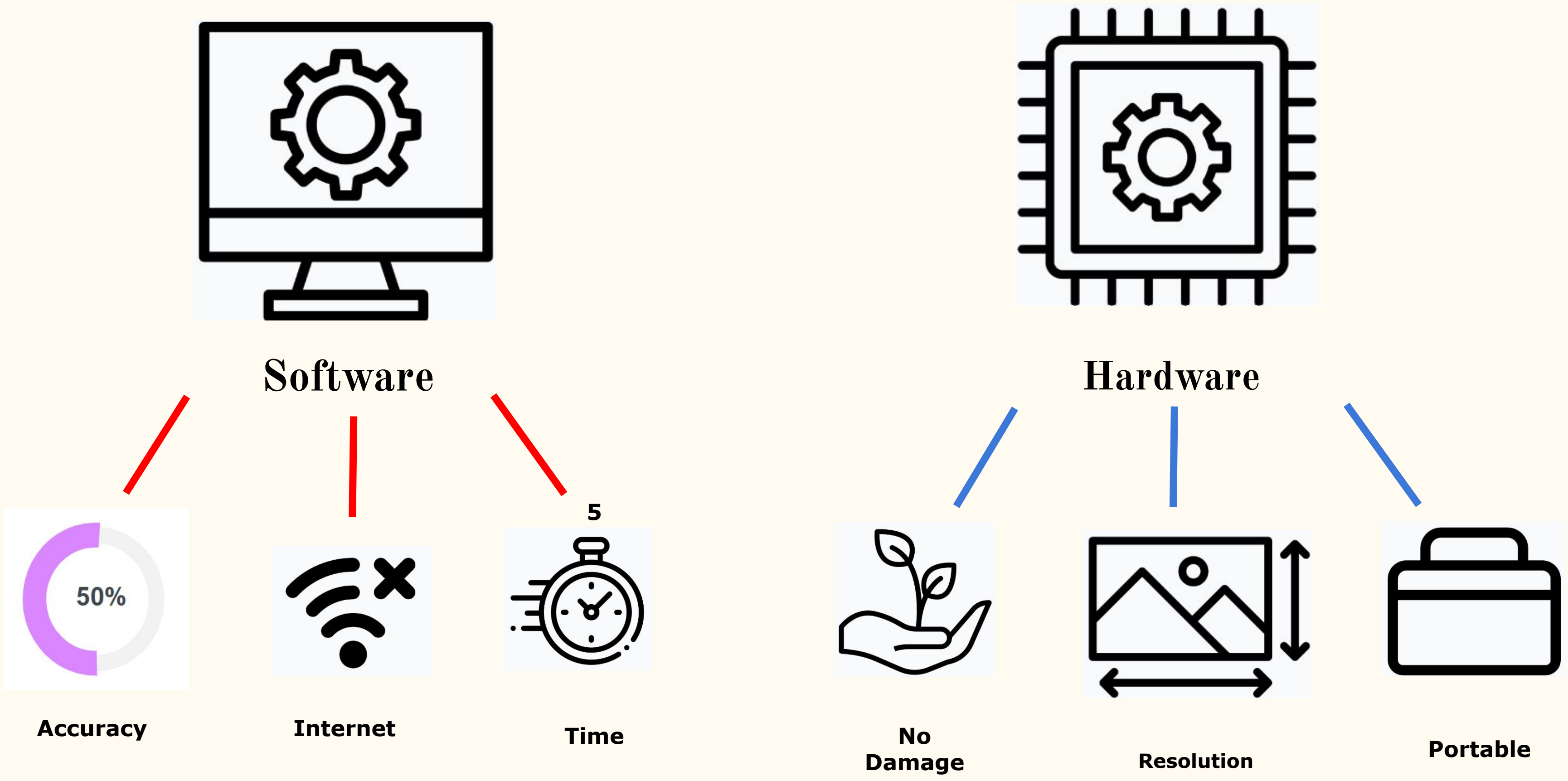
https://www.raspberrypi-spy.co.uk/2015/05/using-an-i2c-enabled-lcd-screen-with-the-raspberry-pi/
https://www.raspberrypi.com/products/raspberry-pi-4-model-b/
https://www.raspberrypi.com/news/new-product-raspberry-pi-high-quality-camera-on-sale-now-at-50/

Resource / Cost Estimate

Item	Cost
Raspberry Pi 4	$45
LCD Screen	$10
Small Motor	$5
Binder Clip	$1
PVC (4ft)	$20
PVC 90 Degree Fitting x2	$5
Colorized Pastic Base Platform (3d Printed)	$25
Handlebar	$10
PVC Swivel Fitting x2	$10
Raspberry Pi Camera v3 (4k)	$40
TOTAL	$171

Functional Requirements

https://www.flaticon.com/kr/free-icons/pictogram/5



Non-Functional Requirements

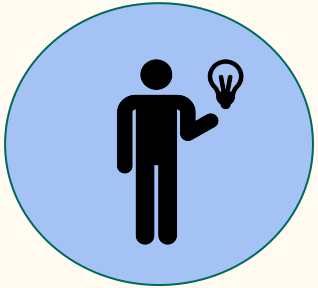
https://www.flaticon.com/kr/free-icons/pictogram/5

Easy to Use

Affordable

No Error

Sturdy



Few Interaction

Undamaged

Constraints (Technical and Otherwise)

- Limited dataset
  - Industry standard is to have over 1500 images with more than 10,000 instances per class
  - We have ~150 images, with ~1500-7500 instances
- Application must fit and run in <5s per image on a Raspberry Pi
  - Limits size of application
  - May limit algorithm choices
- Must be scalable
  - Our device should be able to scale up to large research operations

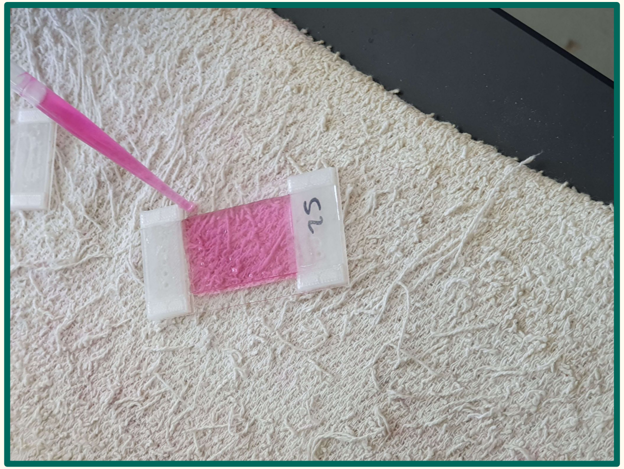
Market Survey

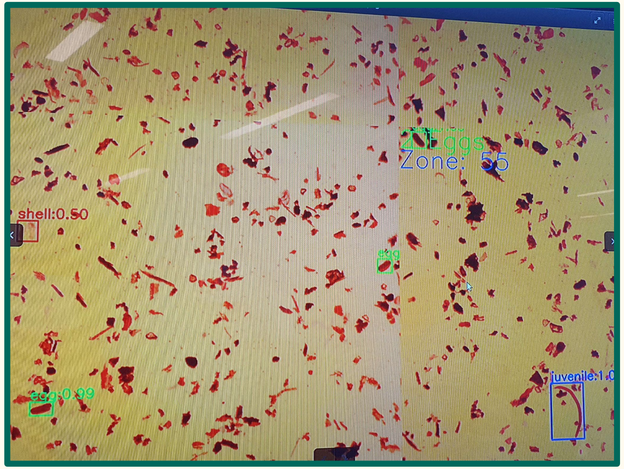
Existing Processes:

- Visual inspection of roots
- Microscope inspection of processed soil samples

Existing Technology:

- Various object detection algorithms
  - YOLO, SSD, R-CNN, for example.
- NVidia Jetson
  - High-powered computing for embedded systems using AI.
  - Key strengths are computing power and pre-trained models





Potential Risks & Mitigation

Task 1 - Developing a Deep Learning Model

Risk	Probability	Mitigation Plan
Available data is not enough to train an accurate algorithm on.	**0.8**	Develop a proof-of-concept model and allow the project administrators to collect more data over time to improve the model.
Labeling tools are incompatible with algorithm implementation.	0.1	N/A
Our algorithm does not provide a sufficient amount of accuracy rating.	0.1	Since we have a relatively low goal accuracy (~50%), even with our limited dataset we should be able to achieve this.

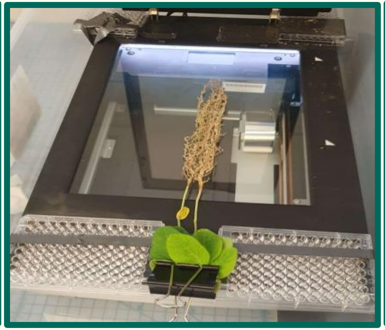
Potential Risks & Mitigation: Task 2

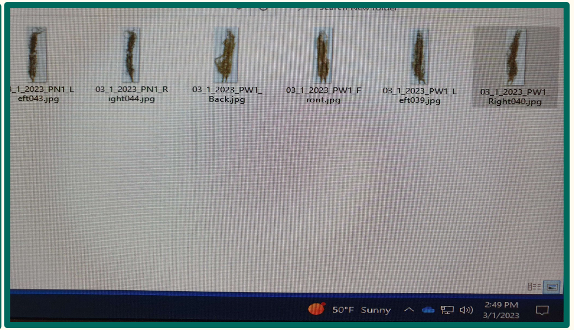
Task 2 - Developing Hardware Prototype

Risk	Probability	Mitigation Plan
Our hardware does not have high enough resolution for machine learning to detect.	**0.5**	Doing market research to find a high-resolution camera at an affordable price This may violate some of our requirements budget wise.

Other Identified Risks

Risk	Probability	Mitigation Plan
Less predictability, especially since no one has a strong background in this area	0.4	Spending time researching machine learning can help us anticipate issues we might have developing and working on an algorithm





Test Plan

Cyst Detector ML Model

- < 50% Error Range of Output
- Produces output in < 5s/image when run on Raspberry Pi
- No fatal errors occur during operation

Proper Communication

- Raspberry Pi
- Camera
- LCD Output Screen

System Testing Plan

Use unit testing to ensure individual units meet the requirements, and to establish baseline data.

Use integration tests to confirm the data flows appropriately from image capture to cyst-count output.

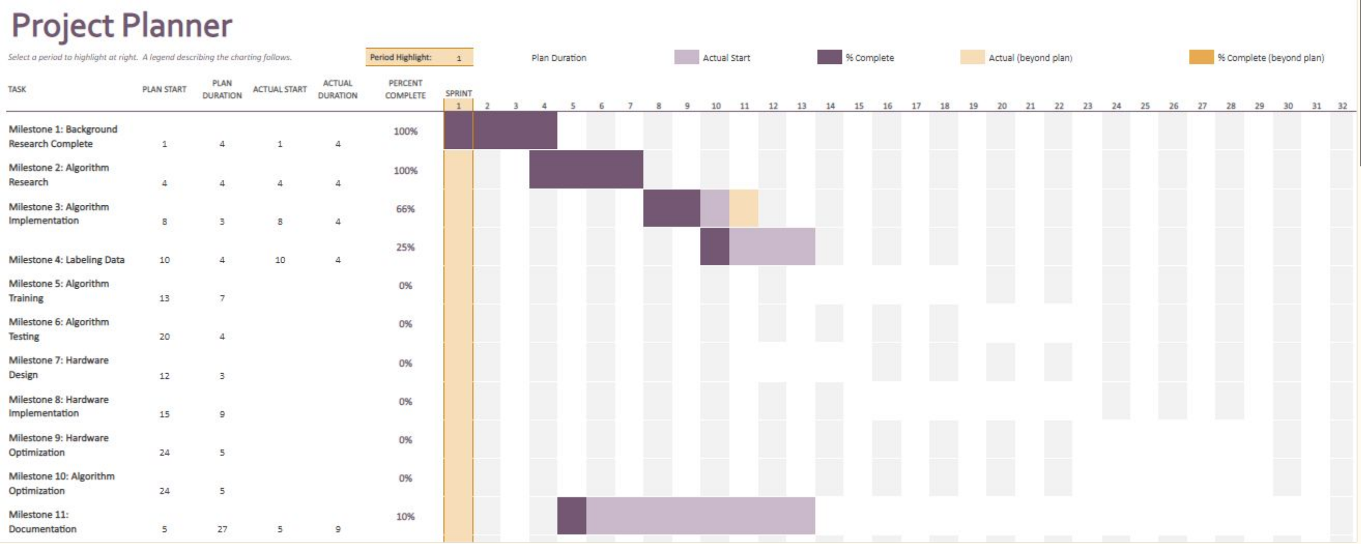
Compare accuracy of the prototype to the baselines established above.

Current Project Status with Respect to Milestones

Completed Milestones
Milestone 1: Background Research Complete
Milestone 2: Algorithm Research
Milestone 3: Algorithm Implementation

Milestones In-Progress
Milestone 4: Labeling Data
Milestone 7: Hardware Design

Expected Completion	Milestones To-Do
Sprint 20	Milestone 5: Algorithm Training
Sprint 24	Milestone 6: Algorithm Testing Milestone 8: Hardware Implementation
Sprint 28	Milestone 9: Hardware Optimization Milestone 10: Algorithm Optimization
Sprint 32	Milestone 11: Documentation



Project Planner

Select a period to highlight at right. A legend describing the charting follows.

Period Highlight: 1 · Plan Duration · Actual Start · % Complete · Actual (beyond plan) · % Complete (beyond plan)

TASK	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Milestone 1: Background Research Complete	1	4	1	4	100%
Milestone 2: Algorithm Research	4	4	4	4	100%
Milestone 3: Algorithm Implementation	8	3	8	4	66%
Milestone 4: Labeling Data	10	4	10	4	25%
Milestone 5: Algorithm Training	13	7			0%
Milestone 6: Algorithm Testing	20	4			0%
Milestone 7: Hardware Design	12	3			0%
Milestone 8: Hardware Implementation	15	9			0%
Milestone 9: Hardware Optimization	24	5			0%
Milestone 10: Algorithm Optimization	24	5			0%
Milestone 11: Documentation	5	27	5	9	10%

Responsibilities / Estimate of Work

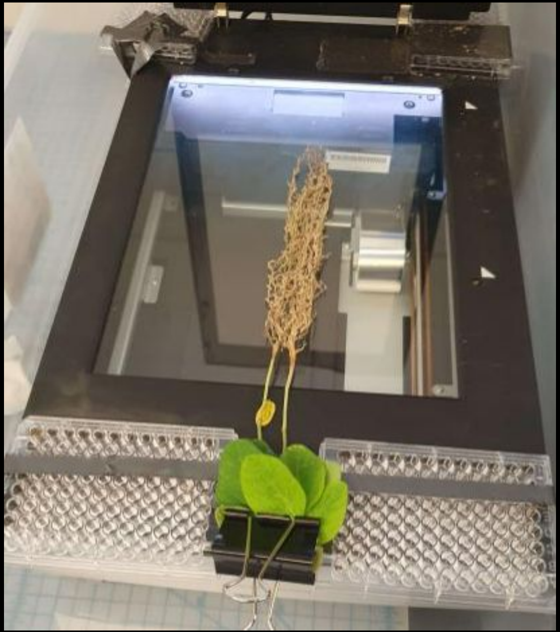
Task: Develop a Deep Learning Model	Person-hours
Research and choose a deep learning algorithm/model	20
Label our existing data	40
Implement our model in our environment	10-15
Train the model on our data	40
Validate & test the model	20
Optimize the code for enhanced improvement	10
(Optionally) implement additional models for comparisons.	60
Software Documentation	50
Total Hours:	***190-255***

Task: Develop a Prototype Soybean Scanner	Person-hours
Set up a controlled environment for image capturing	10
Develop scanner that can scan all sides of the plant	20
Apply the machine learning detector to the scanned sides to accurately count of the parasitic cysts.	15
Optimize the prototype to be user friendly and intuitive.	10
Hardware Documentation	50
Total Hours:	***105***

Total Project Hours:	***295-360***

Plan for next Semester

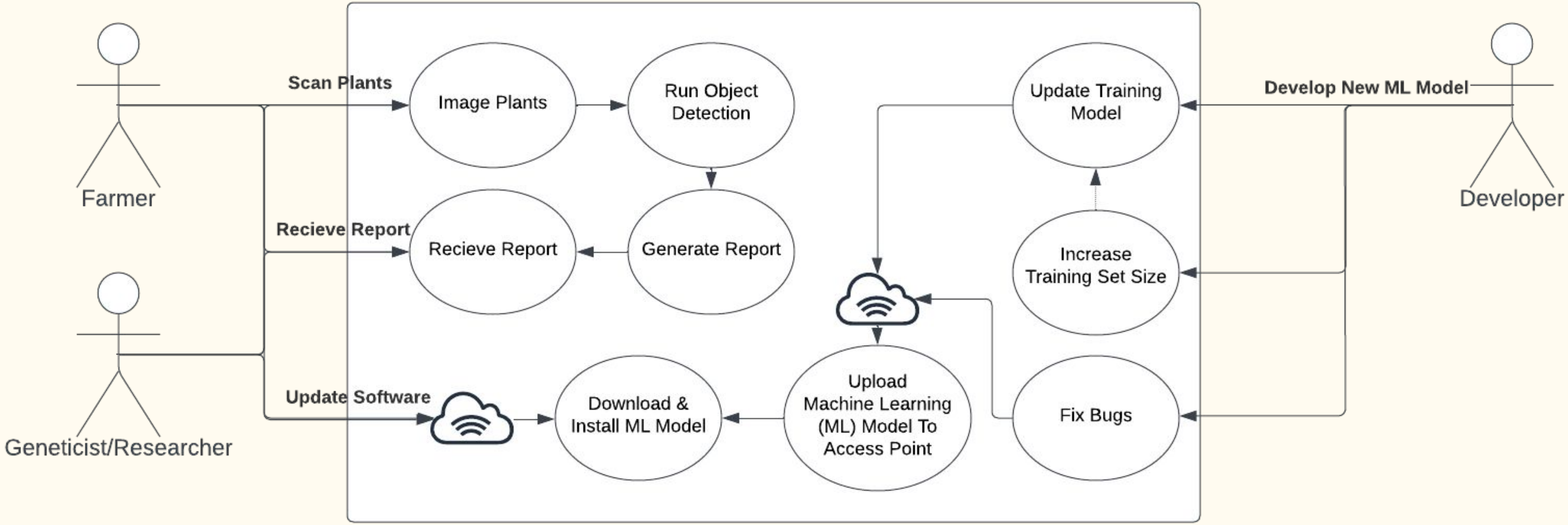
- Summer
  - Data Set Preparation
  - Start training machine learning model
  - Order parts for hardware prototype
- Fall
  - Test and improve accuracy of model
  - Start building prototype
  - Test and optimize prototype




Questions?

Appendix:

Use Case Diagram



Design Exploration

Method	mAP	FPS	batch size	# Boxes	Input resolution
Faster R-CNN (VGG16)	73.2	7	1	~ 6000	$\sim 1000 \times 600$
Fast YOLO	52.7	155	1	98	448×448
YOLO (VGG16)	66.4	21	1	98	448×448
SSD300	74.3	46	1	8732	300×300
SSD512	76.8	19	1	24564	512×512
SSD300	74.3	59	8	8732	300×300
SSD512	76.8	22	8	24564	512×512

Figure depicting accuracy (mAP) and speed (FPS) of the algorithms considered.

Considered 3 high performing object detection algorithms:

- Faster R-CNN
- You Only Look Once (YOLO)
- Single Shot Detector (SDD)

Points of interest for the algorithms:

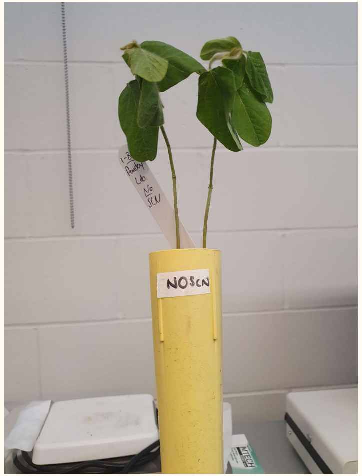
- Classification accuracy
- Algorithm Speed
- Training time
- Region of interest generation
- Small object detection optimization

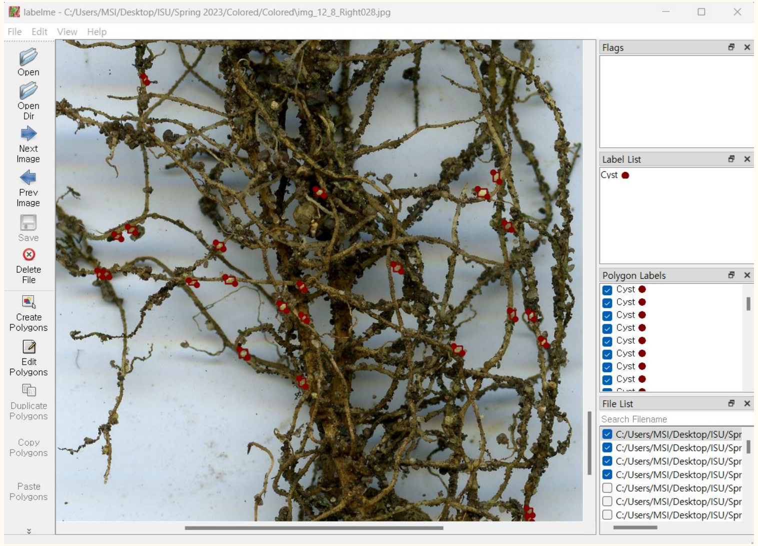
Faster R-CNN vs YOLO vs SSD

- Sifting cysts off the roots and manually counting
- Small object detection algorithms

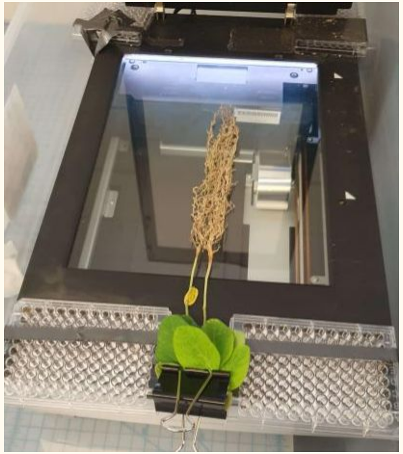
Faster R-CNN	You Only Look Once (YOLO)	Single Shot Detector (SSD)
Uses Region Proposal Network to generate regions containing objects for classification.	Uses Anchor Boxes to generate regions containing objects for classification.	Uses Anchor Box Pyramids to generate regions containing objects for classification.
Uses Convolutional Neural Network to classify objects.	Uses Convolutional Neural Network to classify objects.	Uses Convolutional Neural Network to classify objects.
Algorithm Training takes a considerable amount of time.	Algorithm Training takes a moderate amount of time.	Algorithm Training takes a moderate amount of time.
73% mean Average Precision (mAP)	50-65% mean Average Precision (mAP)	75% mean Average Precision (mAP)
Capable of handling high resolution images.	Capable of handling high resolution images.	Capable of handling high resolution images.

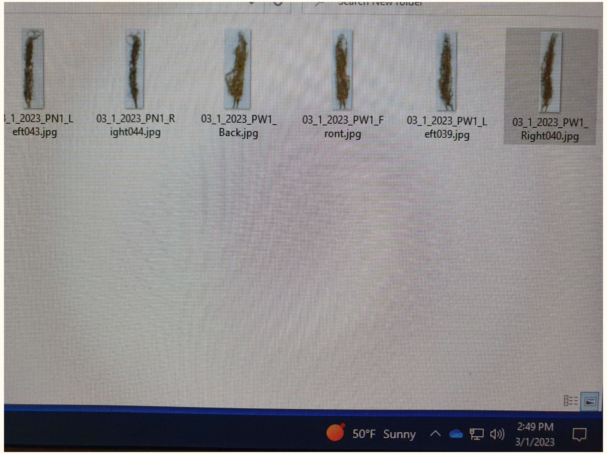
Images: Cyst Nematode Identification Process



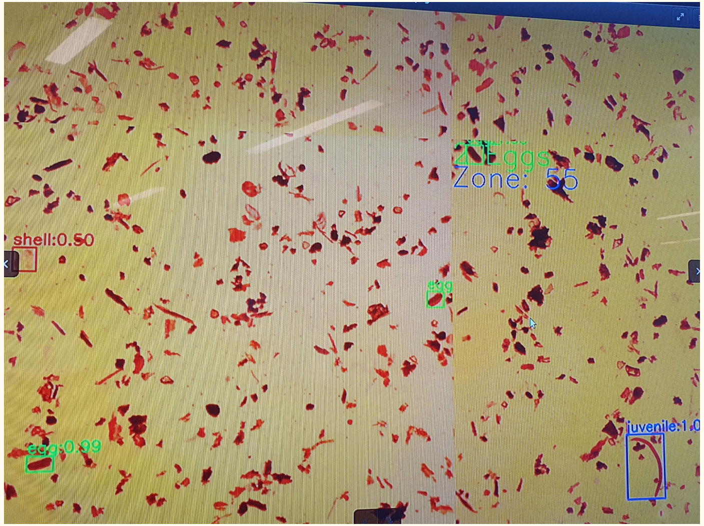


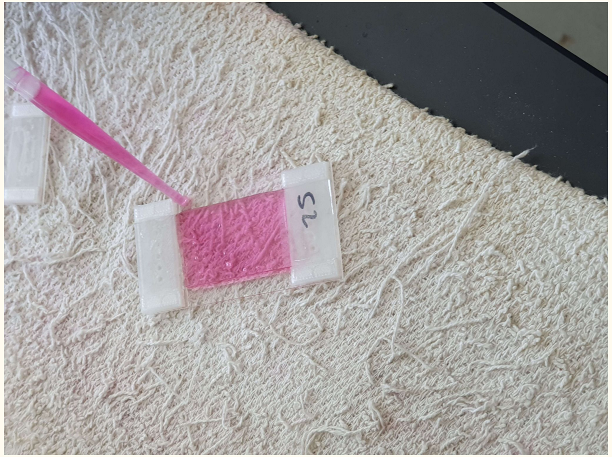
Images: Cyst Nematode Identification Process






DELTA
10" DRILL PRESS


Eggs
Zone: 55
shell:0.50
egg
egg:0.99
juvenile:1.0


25

Mask R-CNN Vs. Faster R-CNN

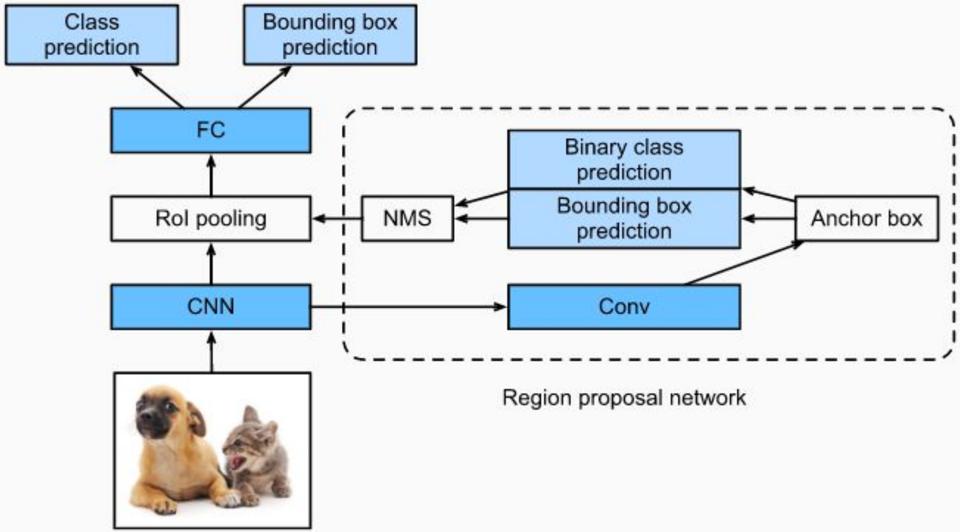


Fig. 14.8.4 The faster R-CNN model.

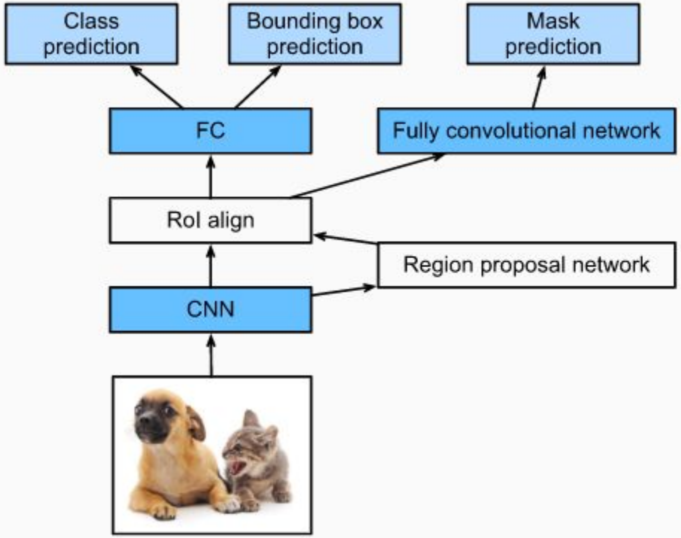
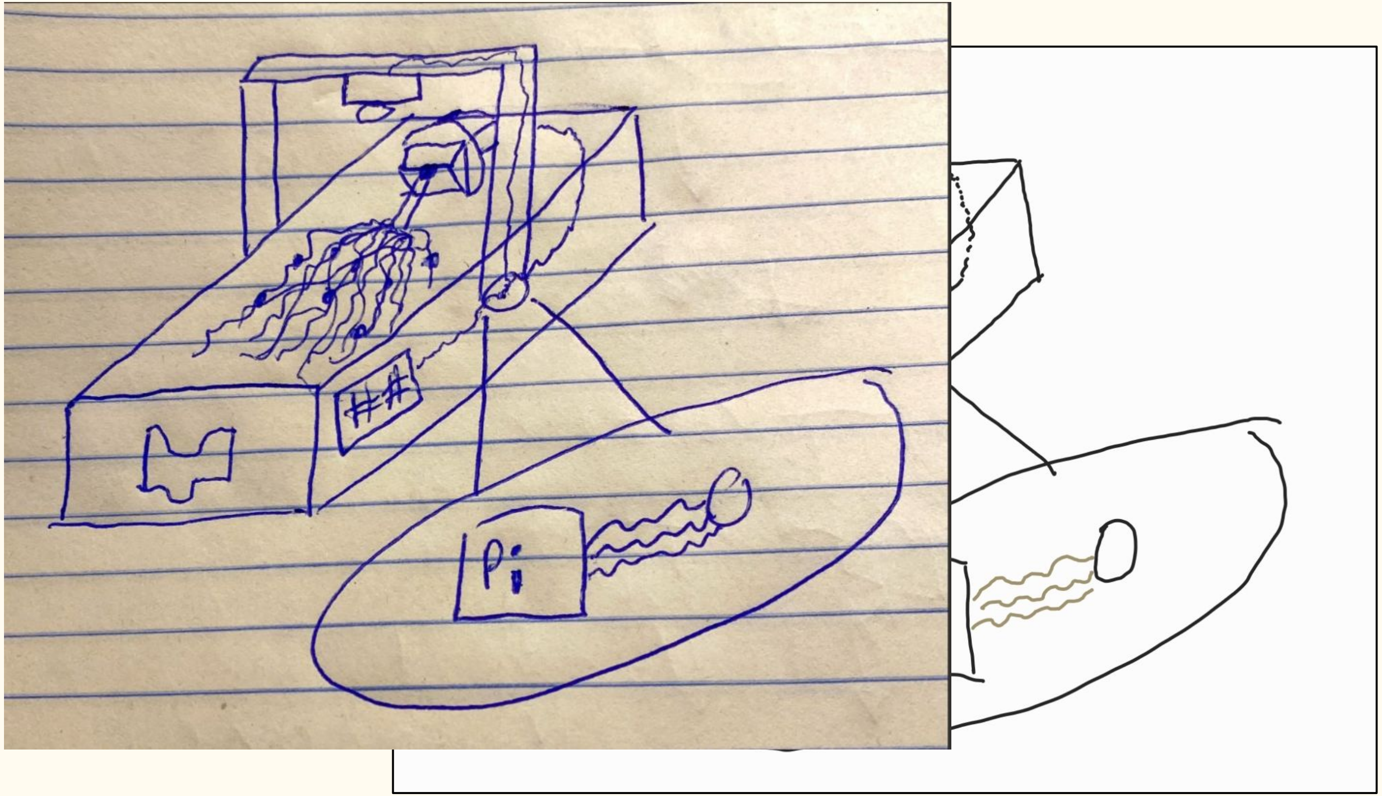


Fig. 14.8.5 The mask R-CNN model.

IEEE Standards

- *IEEE 268-1992:* American National Standard for Metric Practice
- *IEEE/ISO/IEC 32675-2021:* ISO/IEC/IEEE International Standard--Information technology--DevOps--Building reliable and secure systems including application build, package and deployment
- *IEEE/ISO/IEC P24748-6*: ISO/IEC/IEEE Draft Standard - Systems and Software Engineering -- Life Cycle Management
- *IEEE/ISO/IEC 14764-2021:* ISO/IEC/IEEE International Standard - Software engineering - Software life cycle processes Maintenance

Conceptual Sketch



Plan for next Semester

Summer

- Data Set Preparation
- Start training machine learning model
- Order parts for hardware prototype

Fall

- Test and improve accuracy of model
- Start building prototype
- Test and optimize prototype

Test Plan

Tests

- Cyst Detector ML Model
    - < 50% Error Range of Output
    - Produces output in < 5s/image when run on Raspberry Pi
    - No fatal errors occur during operation
- Ensure proper communication between hardware devices
    - Raspberry Pi
    - Camera
    - LCD Output Screen

System Testing Plan

Use unit testing to ensure individual units meet the requirements, and to establish baseline data.

Use integration tests to confirm the data flows appropriately from image capture to cyst-count output.

Compare accuracy of the prototype to the baselines established above.