

INTEGRATING
COMPUTER VISION
AND ROBOTICS FOR
AUTOMATED ROOT
ANALYSIS AND
INOCULATION

Michon Goddijn

231849

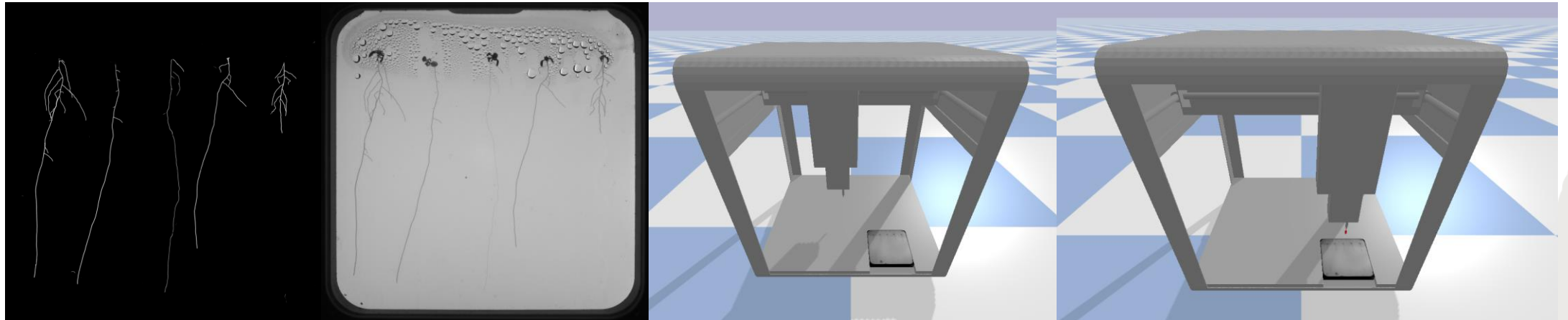


PROBLEM DEFINITION

- **Key Challenges:**
 - Analyzing **root system architectures** in high-throughput environments.
 - Automating **precise inoculations** using liquid-handling robots.
- **Context:**
 - Focus on *Arabidopsis thaliana* grown in Petri dishes.
 - Images show **complex root overlaps**, complicating analysis.
- **My Solution:**
 - Scalable approach for **automated segmentation** and measurement.
 - Integration of **robotic inoculation** for precision and efficiency.

OVERVIEW OF PROPOSED SOLUTION

- **Root Segmentation:** Detect, segment, and measure root lengths.
- **Robotics:** Use PID and RL controllers for precise inoculations.
- **Integration:** Combine computer vision outputs with robotic inputs.



HOW MY SOLUTION WILL HELP THE CLIENT

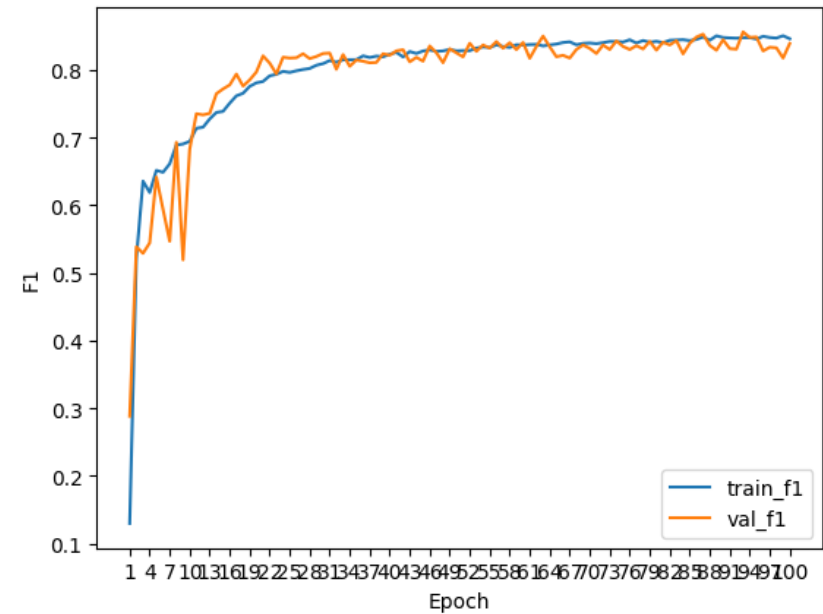
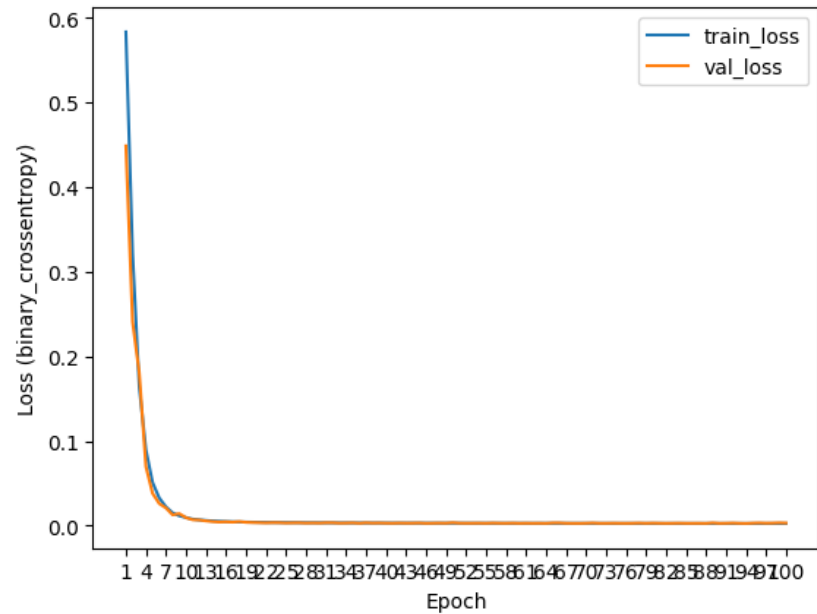
- **High-throughput phenotyping:** Measure thousands of roots efficiently.
- **Precision inoculation:** Deliver microbes to root tips accurately.
- **Scalable experiments:** Enable large-scale plant-microbe studies.



RESULTS & EVALUATION

- **Root Model:** F1 score of 0.856.
- **CV Pipeline:** Achieved sMAPE of 5.12% (public leaderboard) using zone-based root assignment.
- **Controller Performance:** RL and PID controller reached goals with 1mm accuracy.
- **Reinforcement Learning:** Hyperparameter optimization within the group resulted in an RL model that combined stability and speed for efficient task execution.
- **Integration of Computer Vision and Robotics:** Successfully integrated segmentation pipeline with robotics for automated inoculations.

ROOT MODEL PERFORMANCE



Best validation loss: 0.0027233962900936604

Best validation F1: 0.8563873171806335

Learning rate: $1e-4$

Batch size: 32

Epochs: 100

COMPUTER VISION PIPELINE

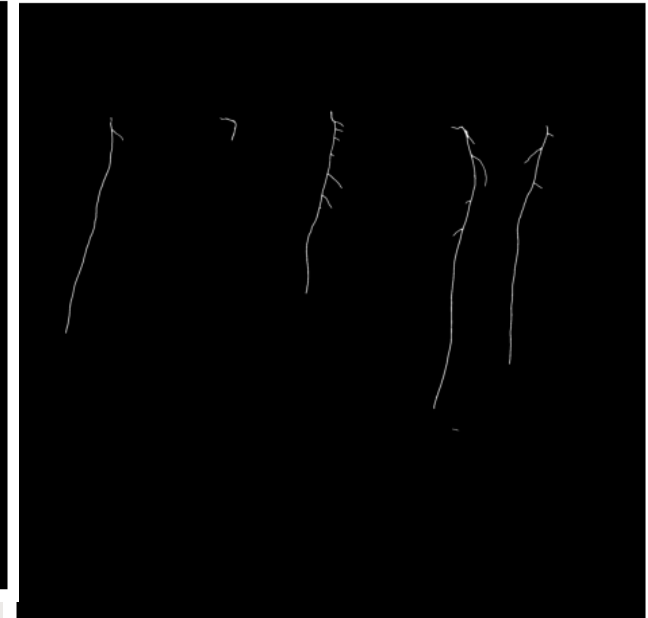
Original Image - test_image_2



Predicted Mask (No Cleaning) - test_image_2



Cleaned Root Mask (Top Excluded) - test_image_2



COMPUTER VISION PIPELINE

Image: test_image_2, Root 1 => Plant 1, Length=1112.68

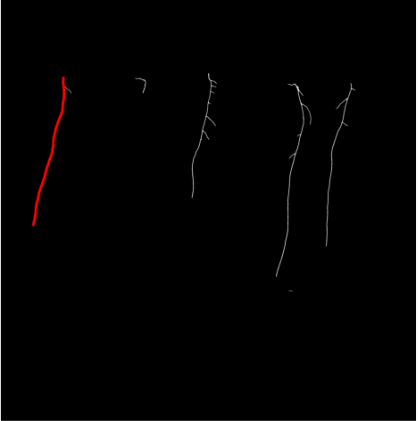


Image: test_image_2, Root 2 => Plant 2, Length=173.23

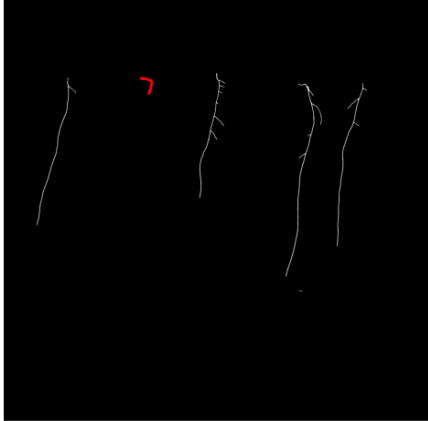


Image: test_image_2, Root 3 => Plant 3, Length=936.90

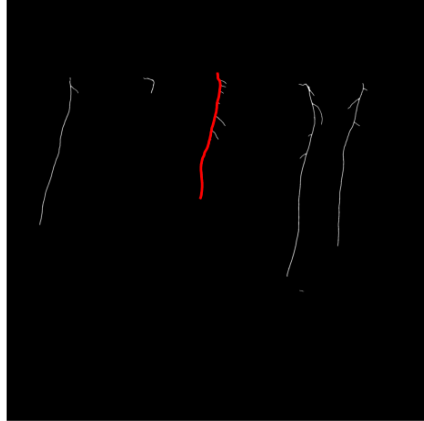


Image: test_image_2, Root 4 => Plant 4, Length=1496.39

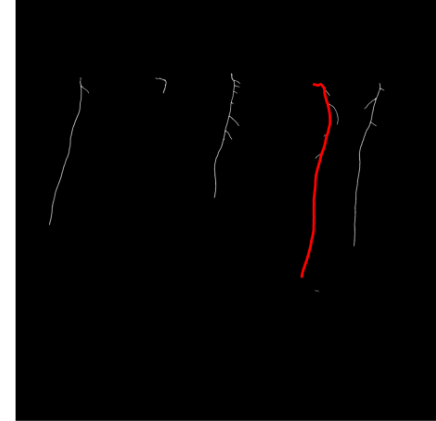
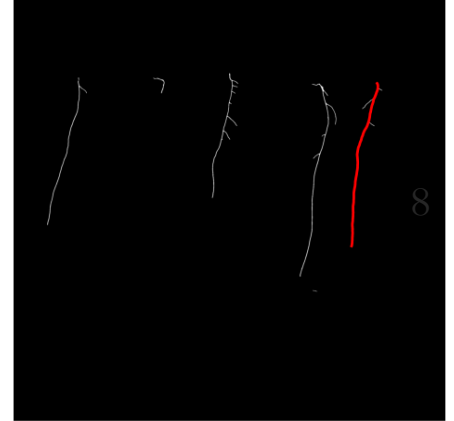
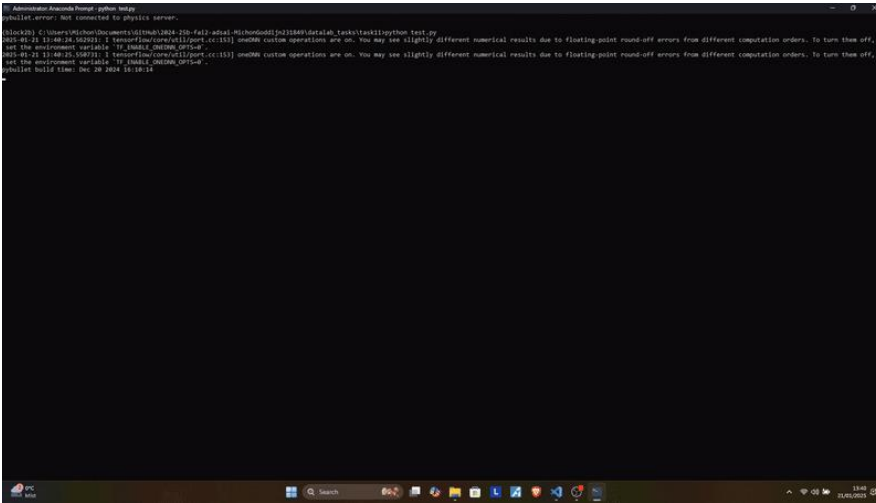


Image: test_image_2, Root 5 => Plant 5, Length=1211.84



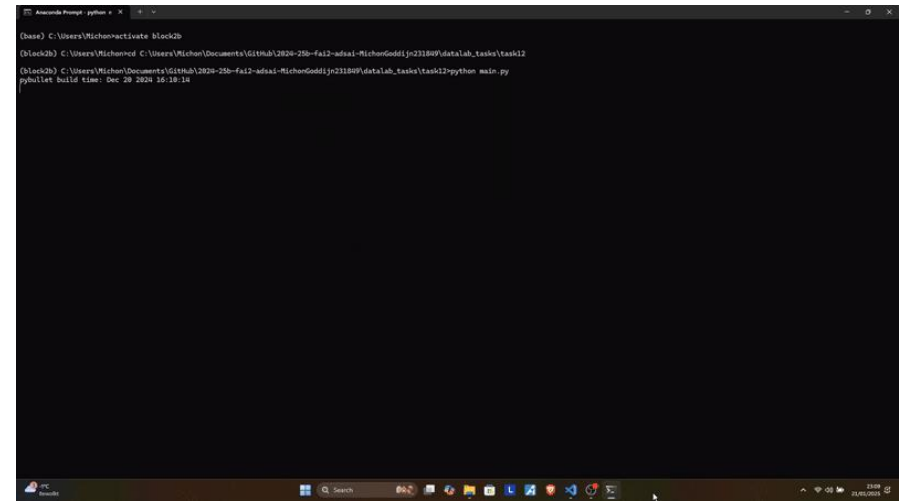
CONTROLLER PERFORMANCE



```
Administrator: Windows PowerShell
pybullet_error: Not connected to physics server.

(block2b) C:\Users\Michon\Documents\GitHub\2024-25b-fall-adai-MichonGoddijn\23809\datalab_tasks\python test.py
Dec 20 21:11:40.25: 62892.1 Temporal error fall (port.cc:151) numNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off,
set the environment variable 'PYBULLET_DISABLE_CUSTOM_NUM_OPERATIONS'.
Dec 20 21:11:40.25: 62893.1 Temporal error cov (port.cc:151) numNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off,
set the environment variable 'PYBULLET_DISABLE_CUSTOM_NUM_OPERATIONS'.
pybullet build time: Dec 20 2024 14:18:14
```

RL CONTROLLER



```
(base) C:\Users\Michon>activate block2b
(block2b) C:\Users\Michon\Documents\GitHub\2024-25b-fall-adai-MichonGoddijn\23809\datalab_tasks\task12
(block2b) C:\Users\Michon\Documents\GitHub\2024-25b-fall-adai-MichonGoddijn\23809\datalab_tasks\task12>python main.py
pybullet build time: Dec 20 2024 14:18:14
```

PID CONTROLLER

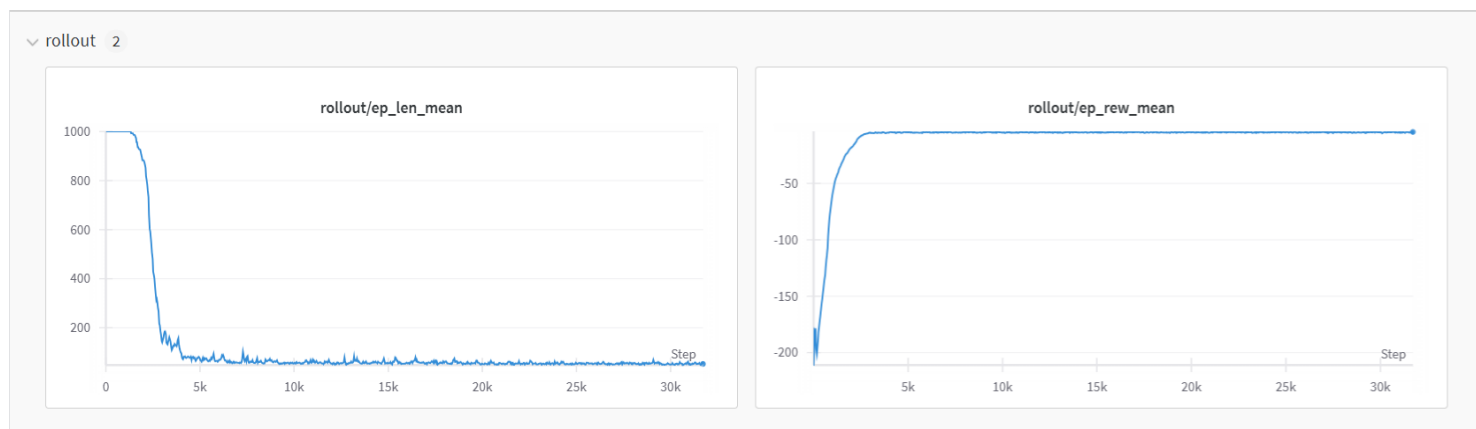
Best Performing Gains:

KP: 20.0

KI: 0.1

KD: 0.005

REINFORCEMENT LEARNING



Group Model

Learning Rate: 0.0001

Batch Size: 64

Steps: 2048

Epochs: 50

Gamma: 0.8

Policy: MlpPolicy

Clip Range: 0.1

Value Coefficient: 0.75

My Best Model

Learning Rate: 0.0001

Batch Size: 32

Steps: 2048

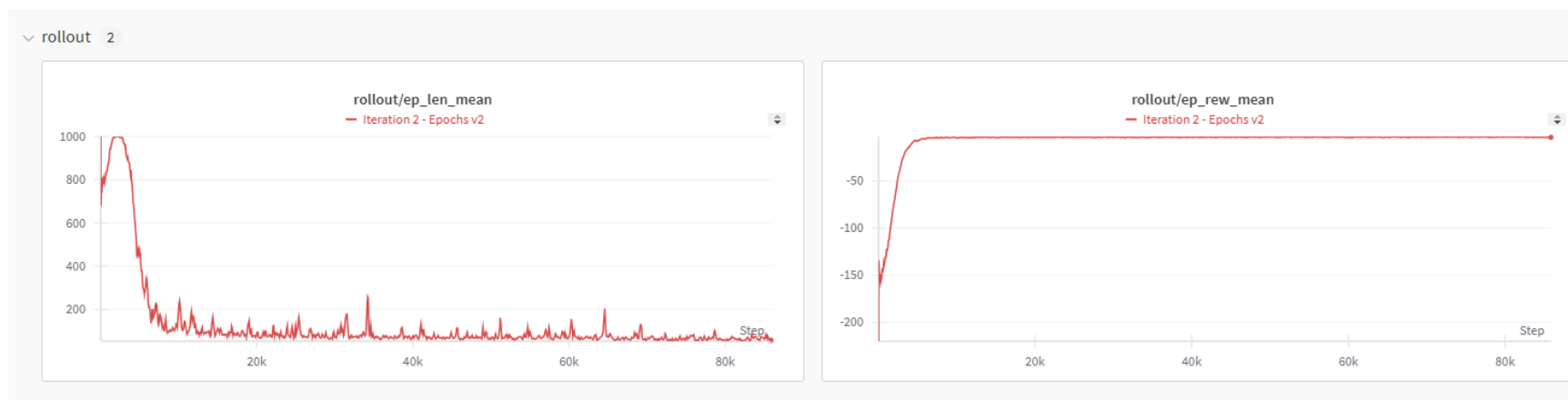
Epochs: 15

Gamma: 0.98

Policy: MlpPolicy

Clip Range: 0.2

Value coefficient: 0.5



INTEGRATION OF COMPUTER VISION AND ROBOTICS RL CONTROLLER PIPELINE

```
Administrator: Anaconda Prompt - python rl_controller_pipeline.py
(block2b) C:\Users\Michon\Documents\GitHub\2024-25b-fall-adlai-MichonGodd\j231849\datalab\task1\python>python rl_controller_pipeline.py
2025-01-21 13:52:07.626677: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off,
set the environment variable 'TF_ENABLE_ONEDNN_OPTS=0'.
2025-01-21 13:52:08.783631: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off,
set the environment variable 'TF_ENABLE_ONEDNN_OPTS=0'.
pybullet build time: Dec 20 2024 16:10:14
```

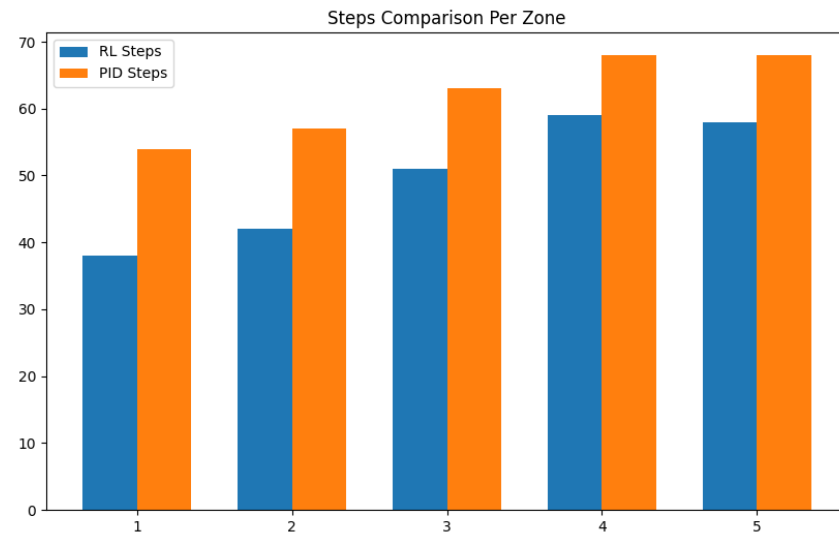
INTEGRATION OF COMPUTER VISION AND ROBOTICS

PID CONTROLLER PIPELINE

```
Anaconda Prompt - python 3
(base) C:\Users\Michon>activate block2b
(block2b) C:\Users\Michon>cd C:\Users\Michon\Documents\GitHub\2024-25b-fai2-adsai-MichonGoddijn231849\datlab_tasks\task13
(block2b) C:\Users\Michon\Documents\GitHub\2024-25b-fai2-adsai-MichonGoddijn231849\datlab_tasks\task13>python pid_controller_pipeline.py
pybullet build time: Dec 20 2024 16:10:14
2025-01-21 22:37:30.415044: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off, set the environment variable 'TF_ENABLE_ONEDNN_OPTS=0'.
2025-01-21 22:37:31.443229: I tensorflow/core/util/port.cc:153] oneDNN custom operations are on. You may see slightly different numerical results due to floating-point round-off errors from different computation orders. To turn them off, set the environment variable 'TF_ENABLE_ONEDNN_OPTS=0'.
```

INTEGRATION OF COMPUTER VISION AND ROBOTICS

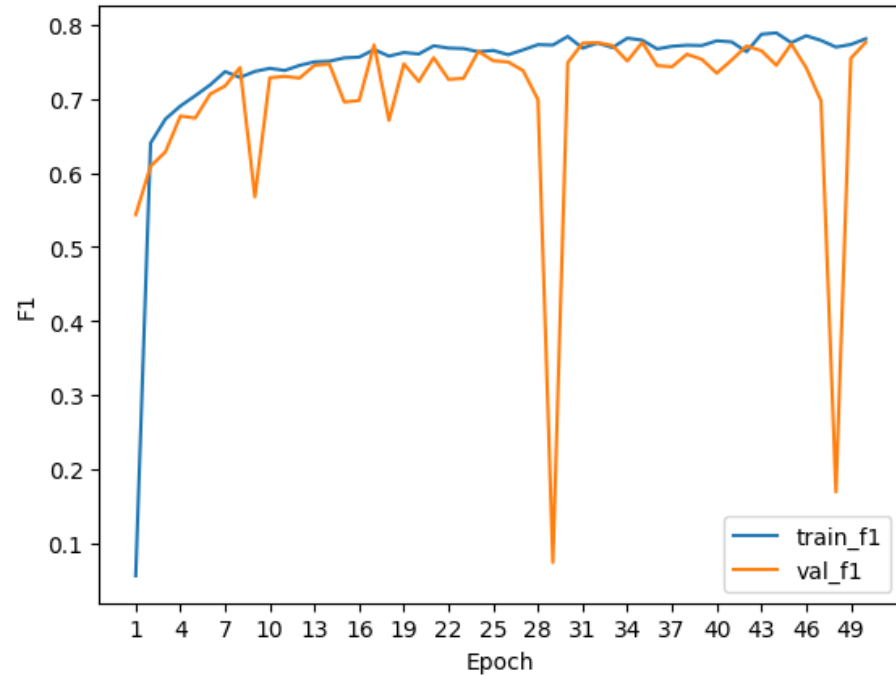
BENCHMARK RESULTS



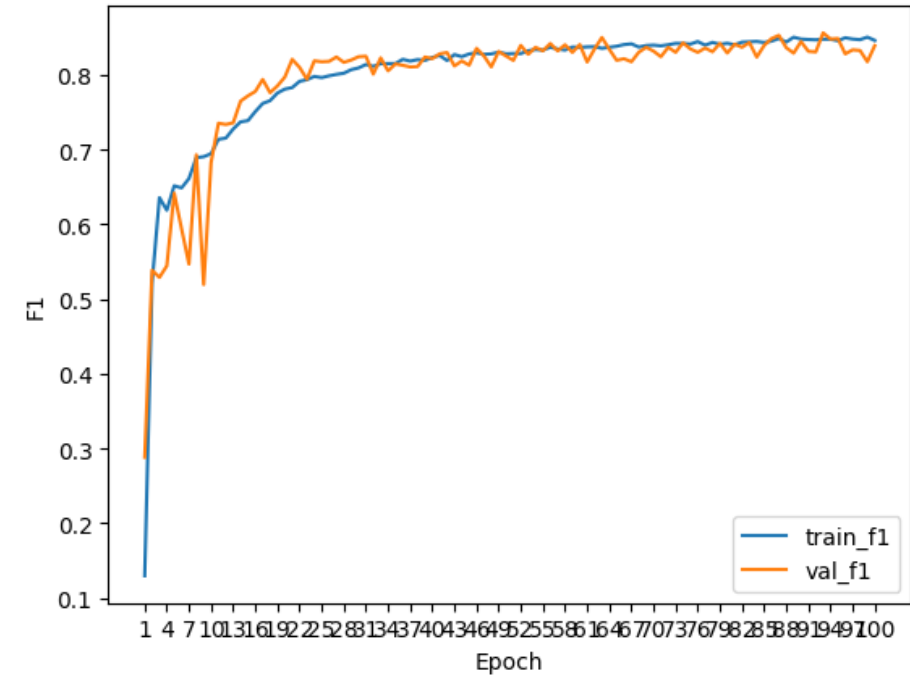
```
=== Benchmark Results (Per Zone) ===
zone  rl_steps  rl_distance  rl_time_sec  rl_success  pid_steps  pid_distance  pid_time_sec  pid_success
0     1         38      0.000933    5.188570    True      54      0.000875    6.089450    True
1     2         42      0.000856    5.587958    True      57      0.000873    6.879905    True
2     3         51      0.000834    6.538404    True      63      0.000834    7.511728    True
3     4         59      0.000728    7.372252    True      68      0.000831    8.043357    True
4     5         58      0.000990    7.254793    True      68      0.000831    8.039656    True

=== Summary ===
Avg RL Time: 6.388 s
Avg PID Time: 7.313 s
Avg RL Steps: 49.6
Avg PID Steps: 62.0
RL Successes: 5 / 5
PID Successes: 5 / 5
```

ERROR ANALYSIS & ITERATIONS



- Batch size: 16
- Epochs: 50



- Batch size: 32
- Epochs: 100
- Learning rate: 1e-4

Root Model: Validation F1 from 0.77 to 0.85

ERROR ANALYSIS & ITERATIONS

Image: test_image_16, Root 2 => Plant 2, Length=422.04

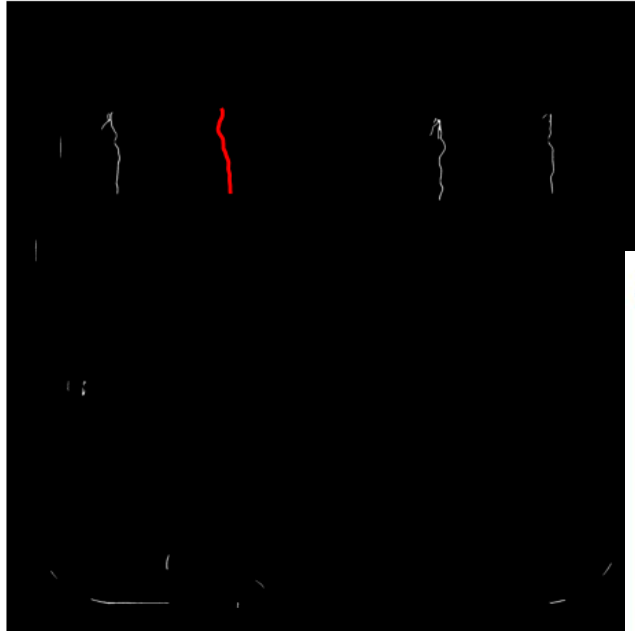
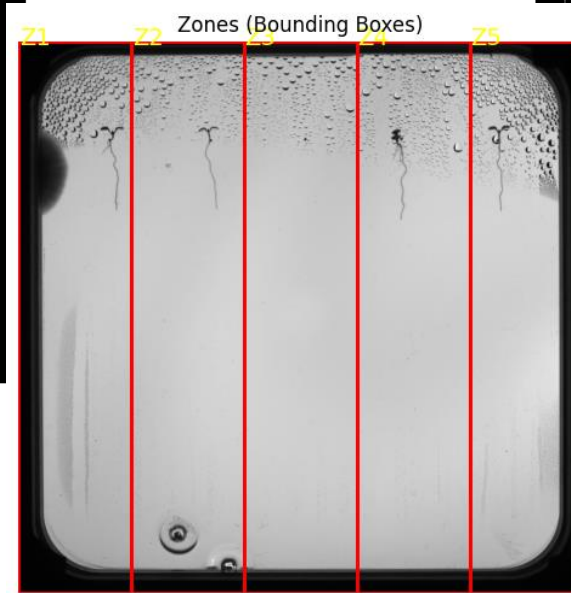
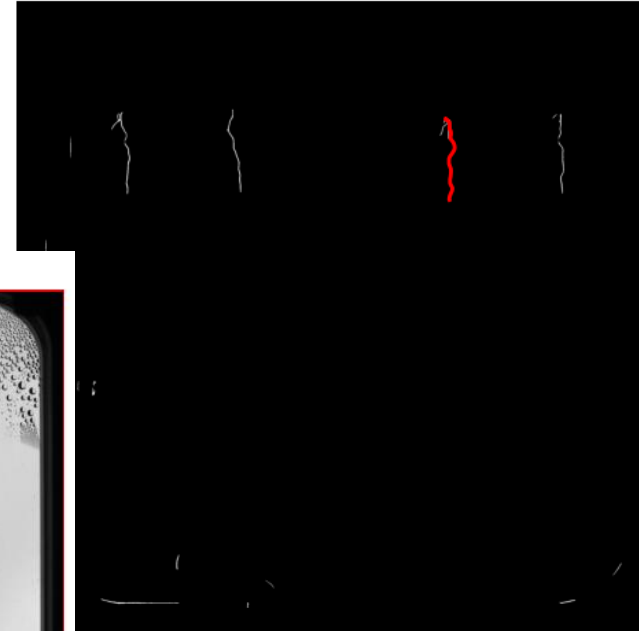


Image: test_image_16, Root 3 => Plant 4, Length=437.81



CV Pipeline: Zone mapping improved sMAPE from from 30.007% to 5.12% (public leaderboard)

ERROR ANALYSIS & ITERATIONS

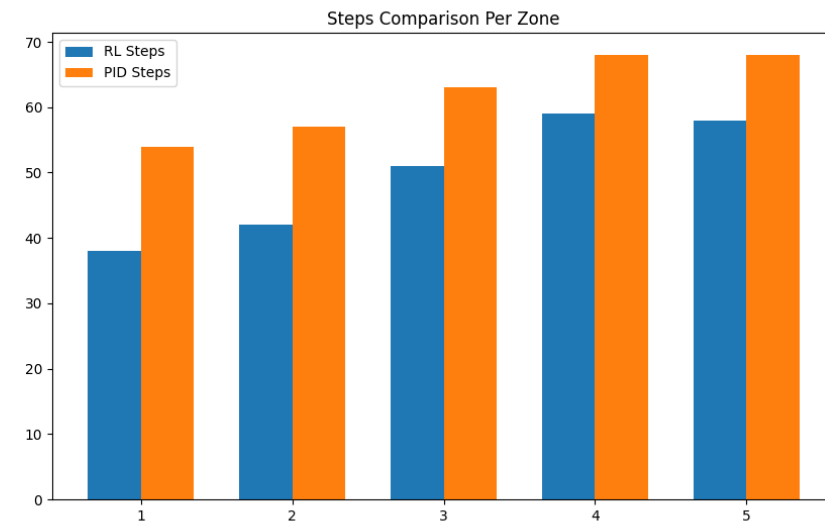
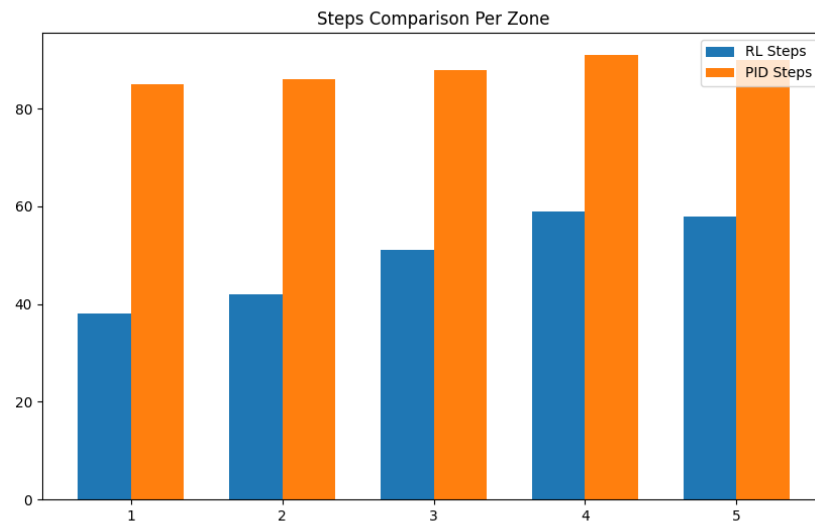
```
Step 104: Current position: [ 0.1714 -0.1296 0.2523], Distance to goal: 0.001239 m  
PID outputs: vx=0.015, vy=-0.050, vz=0.018  
Action: [ 0.01471868 -0.0497707 0.0177053 ]  
Step 105: Current position: [ 0.1715 -0.1298 0.2524], Distance to goal: 0.001024 m  
PID outputs: vx=0.014, vy=-0.048, vz=0.017  
Action: [ 0.013709 -0.04777577 0.01670529]  
Step 106: Current position: [ 0.1715 -0.13 0.2524], Distance to goal: 0.000824 m  
PID outputs: vx=0.014, vy=-0.046, vz=0.017  
Action: [ 0.01371933 -0.04577999 0.01671545]  
Goal reached within 1 mm accuracy at step 106, distance: 0.000824 m
```



```
Step 52: Current position: [0.0753 0.0569 0.2165], Distance to goal: 0.001261 m  
PID outputs: vx=0.001, vy=-0.009, vz=0.032  
Action: [ 0.00107362 -0.00946317 0.03173855]  
Step 53: Current position: [0.0753 0.0568 0.2166], Distance to goal: 0.001139 m  
PID outputs: vx=0.001, vy=-0.007, vz=0.030  
Action: [ 0.00107385 -0.00745452 0.02974405]  
Step 54: Current position: [0.0753 0.0568 0.2168], Distance to goal: 0.000946 m  
PID outputs: vx=0.001, vy=-0.007, vz=0.026  
Action: [ 0.00107408 -0.00746583 0.02573852]  
Goal reached within 1 mm accuracy at step 54, distance: 0.000946 m
```

PID Controller and Pipeline: Speed improved by around 30% - 40% increasing KP from 10 to 20.

ERROR ANALYSIS & ITERATIONS



PID Controller and Pipeline: Speed improved by around 30% - 40% increasing KP from 10 to 20.

ASSUMPTIONS

- Assumed Petri dishes were square, ignoring corner curvatures, to simplify edge detection and cropping.
- Relied on roots being segmented from left to right within predefined zones for plant assignment.
- Assumed consistent lighting and background conditions across datasets to minimize preprocessing variability.
- Assumed a linear transformation could effectively map pixel coordinates to robot positions without accounting for complex distortions.
- Assumed all input images would contain valid, distinguishable roots to prevent errors in segmentation and measurement.

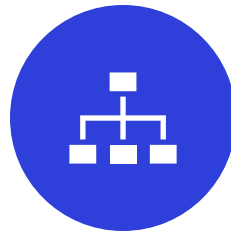
LIMITATIONS



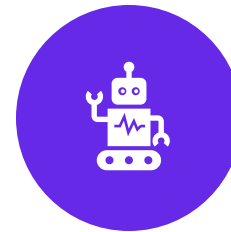
GAPS IN ROOT
PREDICTIONS REDUCE
SEGMENTATION AND
TIP ACCURACY.



STRUGGLES WITH VERY
SMALL ROOTS AND
COMPLEX OVERLAPS.



COMPLEX OVERLAPS
REMAIN
CHALLENGING TO
SEGMENT.



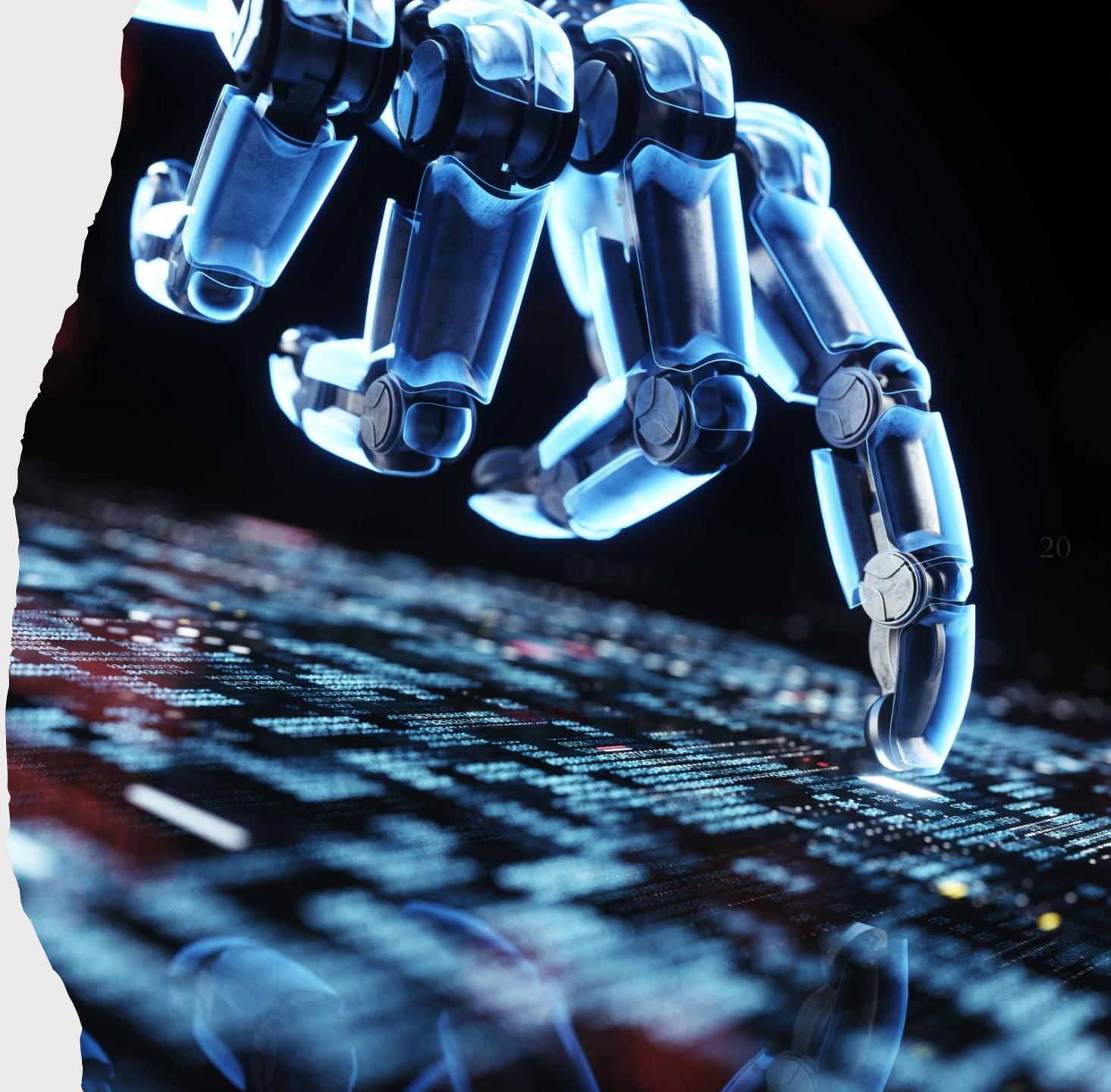
ROBOT ARM DOESN'T
FULLY DESCEND
DURING
INOCULATIONS.



PID CONTROLLER IS
SLOWER COMPARED
TO RL IN REAL-TIME
SCENARIOS.

NEXT STEPS

- Data Augmentation
- Hybrid Controllers
- Robotics Refinement
- Advanced Segmentation Models
- Improved Root Tip Localization
- Error Analysis Integration



SUMMARY



We developed an integrated vision and robotics solution to address challenges in high-throughput phenotyping and precision inoculation of *Arabidopsis thaliana*.



The root segmentation pipeline achieved a validation F1 score of **0.856**, effectively distinguishing roots even in complex images with overlaps.



Our iterative improvements, such as zone-based segmentation, reduced SMAPE from **30.007%** to **5.12%**, significantly enhancing root-to-plant assignment accuracy.



The robotics system successfully integrated PID and RL controllers, achieving millimeter precision while automating inoculation tasks.



Despite limitations like gaps in root predictions and challenges with very small roots, the pipeline provides a robust foundation for further refinement.



Next steps focus on improving root tip localization, handling small root detection, and enhancing robotic inoculation accuracy.