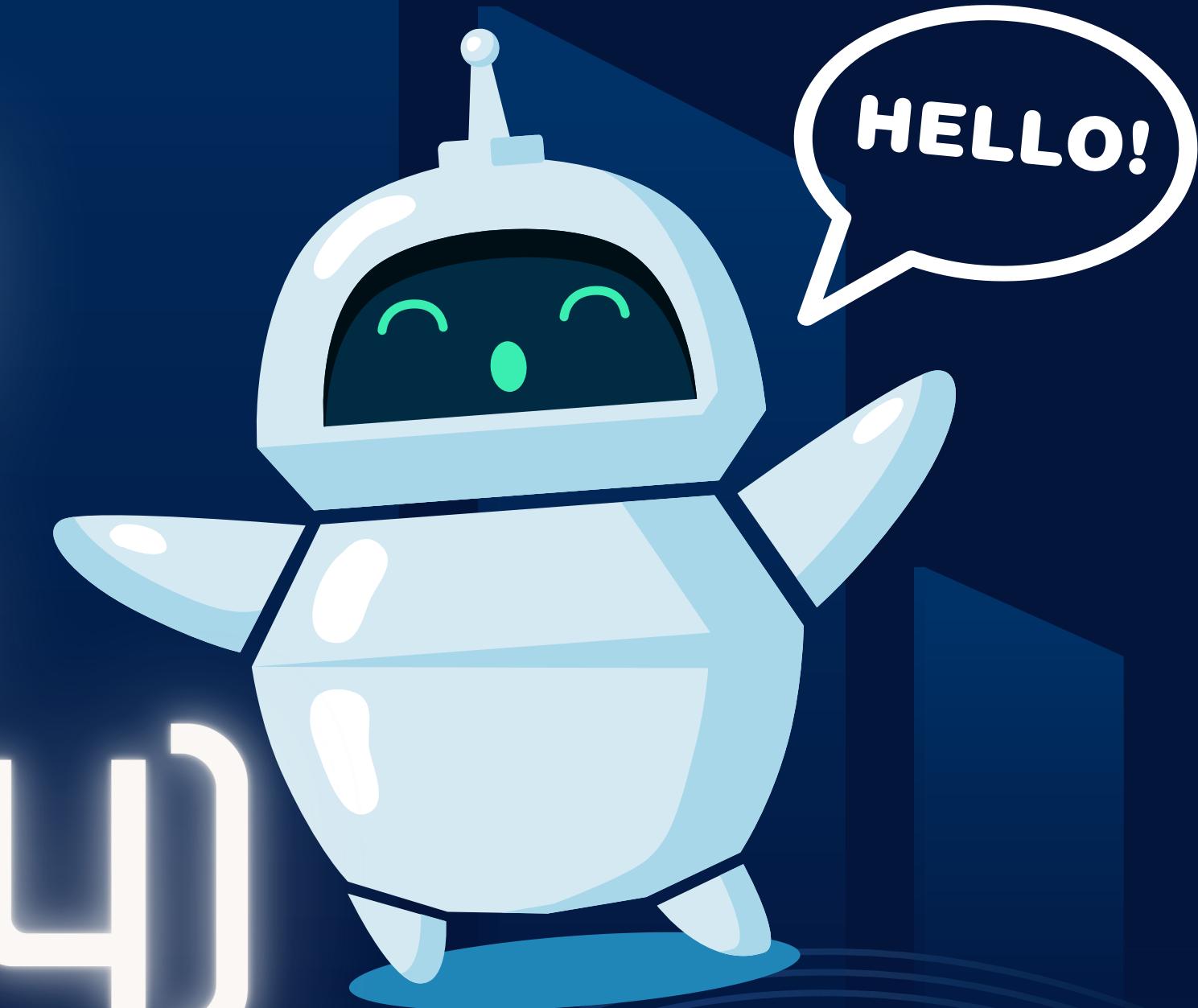


Presented By  
Chiv Pheng Phearin

# Digit Card Detection Robot (0-4)



# Table of Content

- Project Overview
- Digit to Action Mapping
- Dataset collection
- Labeling Samples
- Final Dataset Summary
- Model Architecture (YOLOv8n)
- Training Configuration
- Training Results
- Confusion Matrix
- ONNX Deployment on Raspberry Pi 4
- Challenges & Lessons Learned
- Demo Video



# Project Overview

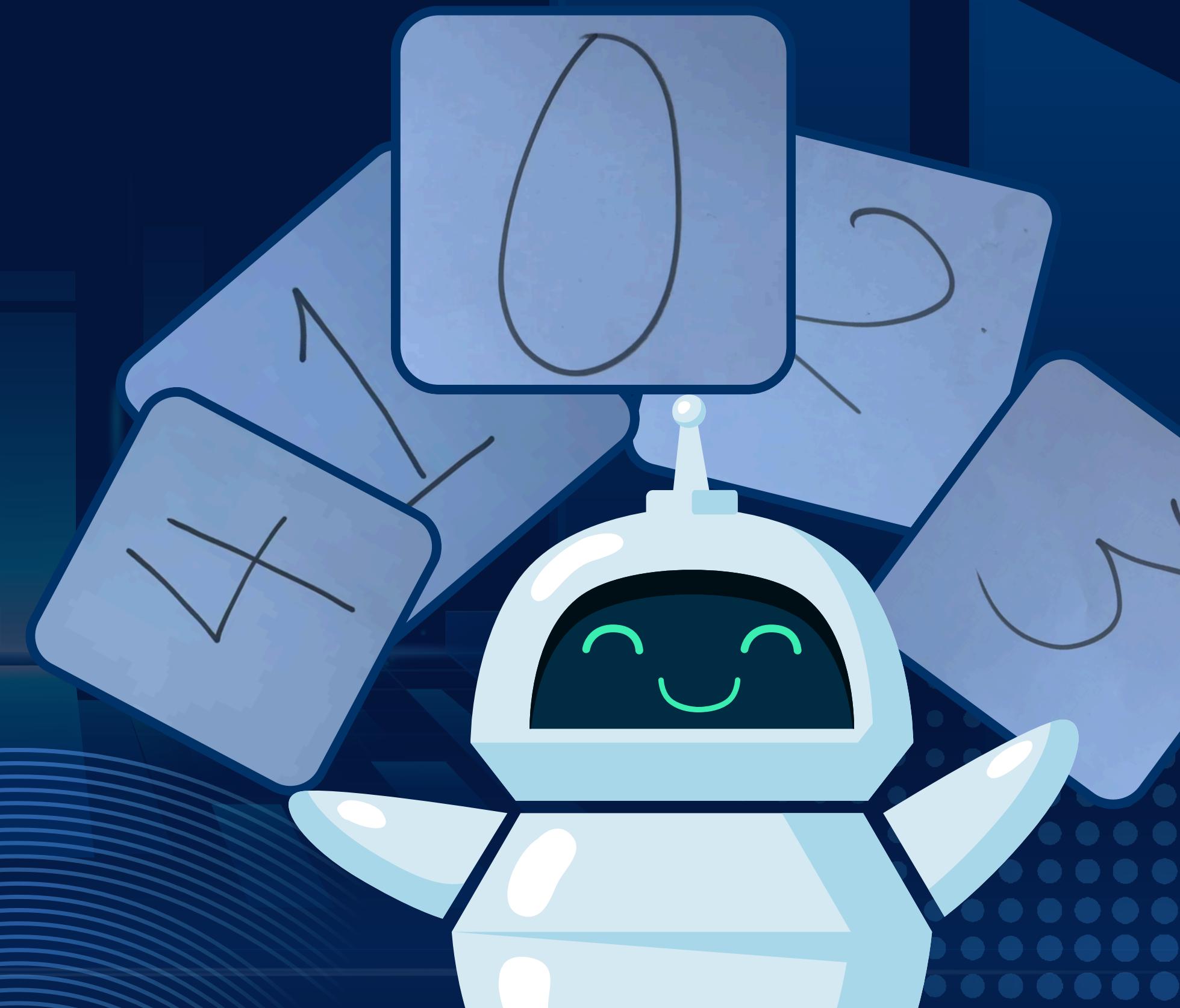
- The goal of this project is to detect printed digits (0–4).
- A custom dataset was collected and labeled.
- Model: YOLOv8n for detection.
- Model exported to ONNX for real-time inference on Raspberry Pi 4.
- Detection output controls robot actions (move, turn, stop).

# Digit to Action Mapping

Digits	Robot Action
0	Stop
1	Move Forward (Slow)
2	Move Forward (Medium)
3	Move Forward (Fast)
4	Turn Left (Small)

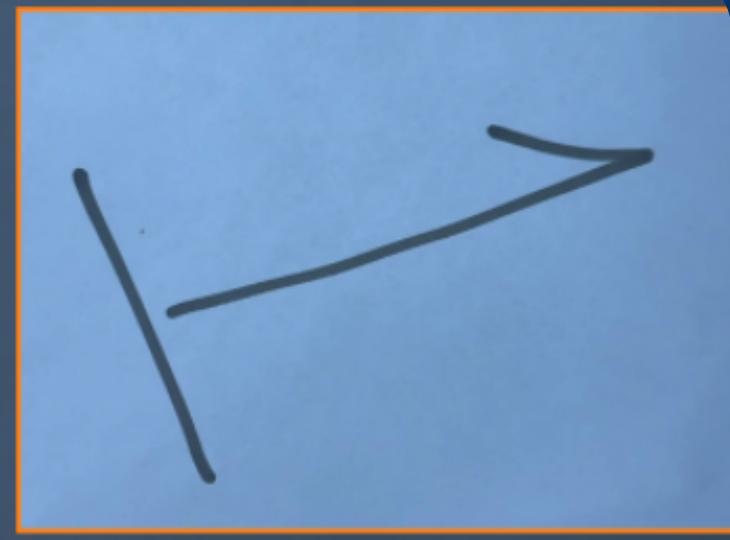
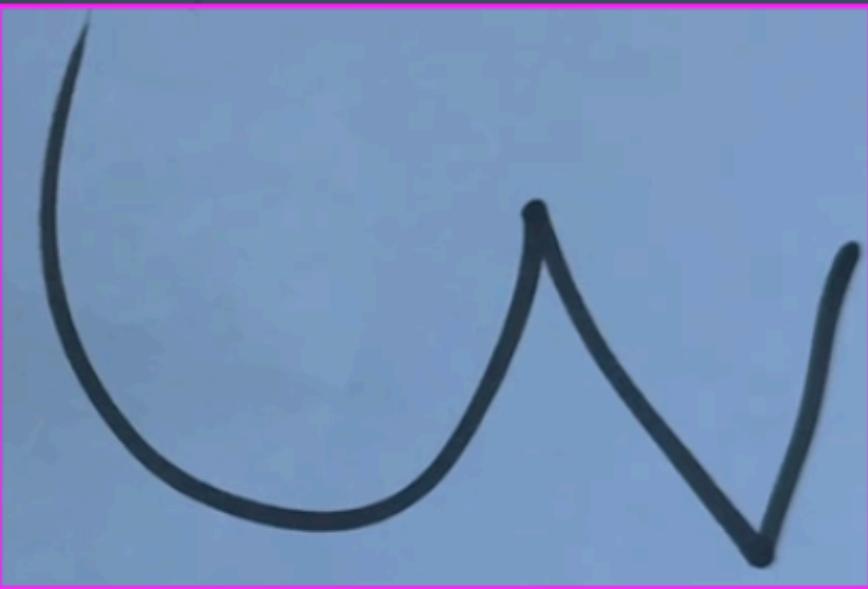
# Dataset Collection

- Collected 1,828 raw images
- Digits captured in many lighting conditions
- Different distances and angles
- Added background images for realism



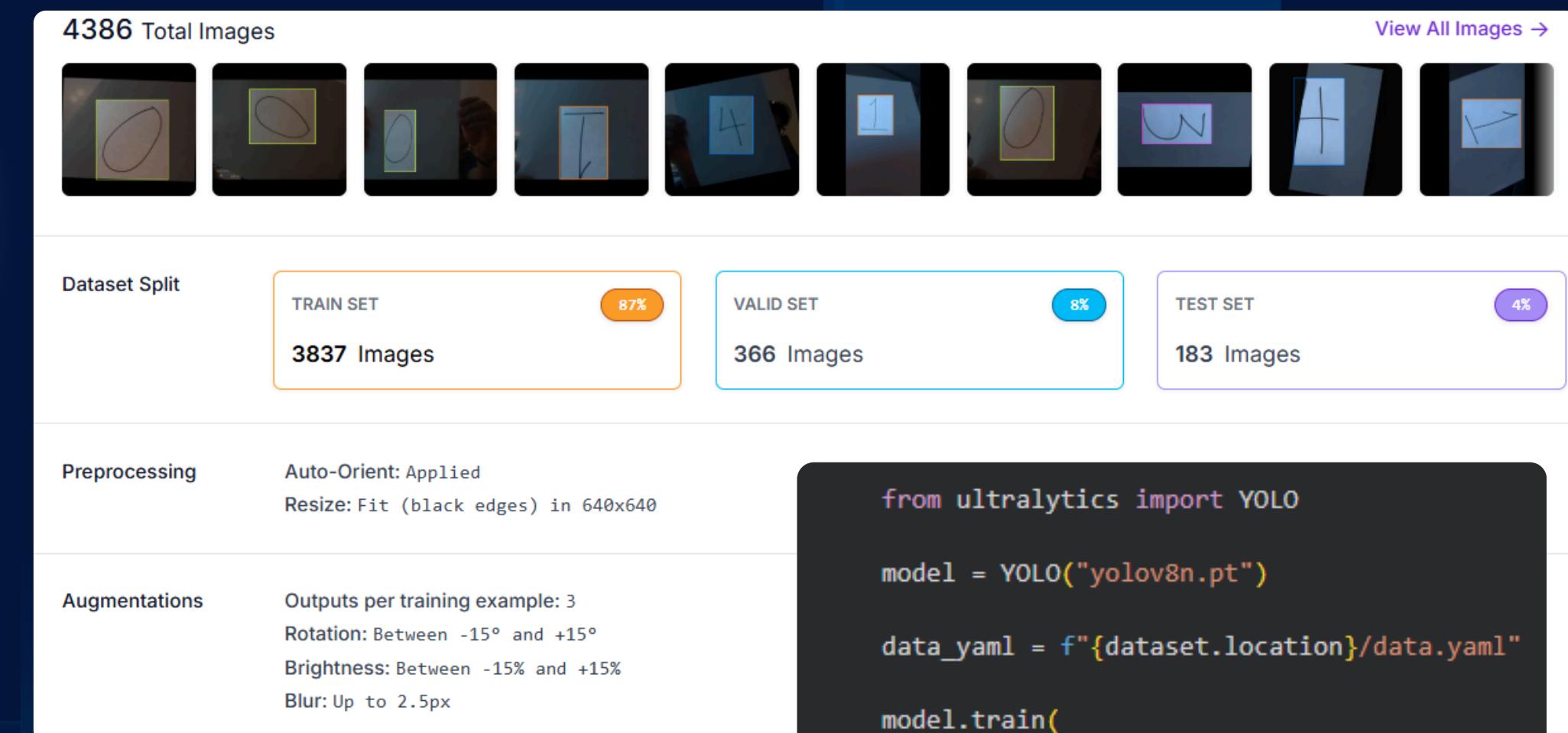
# Labeling Dataset

- All digit regions were labeled using Roboflow
- YOLOv8 annotation format used
- Classes: Digit 0, Digit 1, Digit 2, Digit 3, Digit 4



# Final Dataset Summary

- After augmentation → 4,386 images
- Train: 87%
- Validation: 8%
- Test: 4%
- Image size: 640×640
- Batch size: 16
- Epochs (for evaluation): 10



```
from ultralytics import YOLO  
  
model = YOLO("yolov8n.pt")  
  
data_yaml = f"{dataset.location}/data.yaml"  
  
model.train(  
    data=data_yaml,  
    epochs=10,  
    imgsz=640,  
    batch=16,  
    patience=20  
)
```

Show hidden output



# Model Architecture (YOLOv8n)



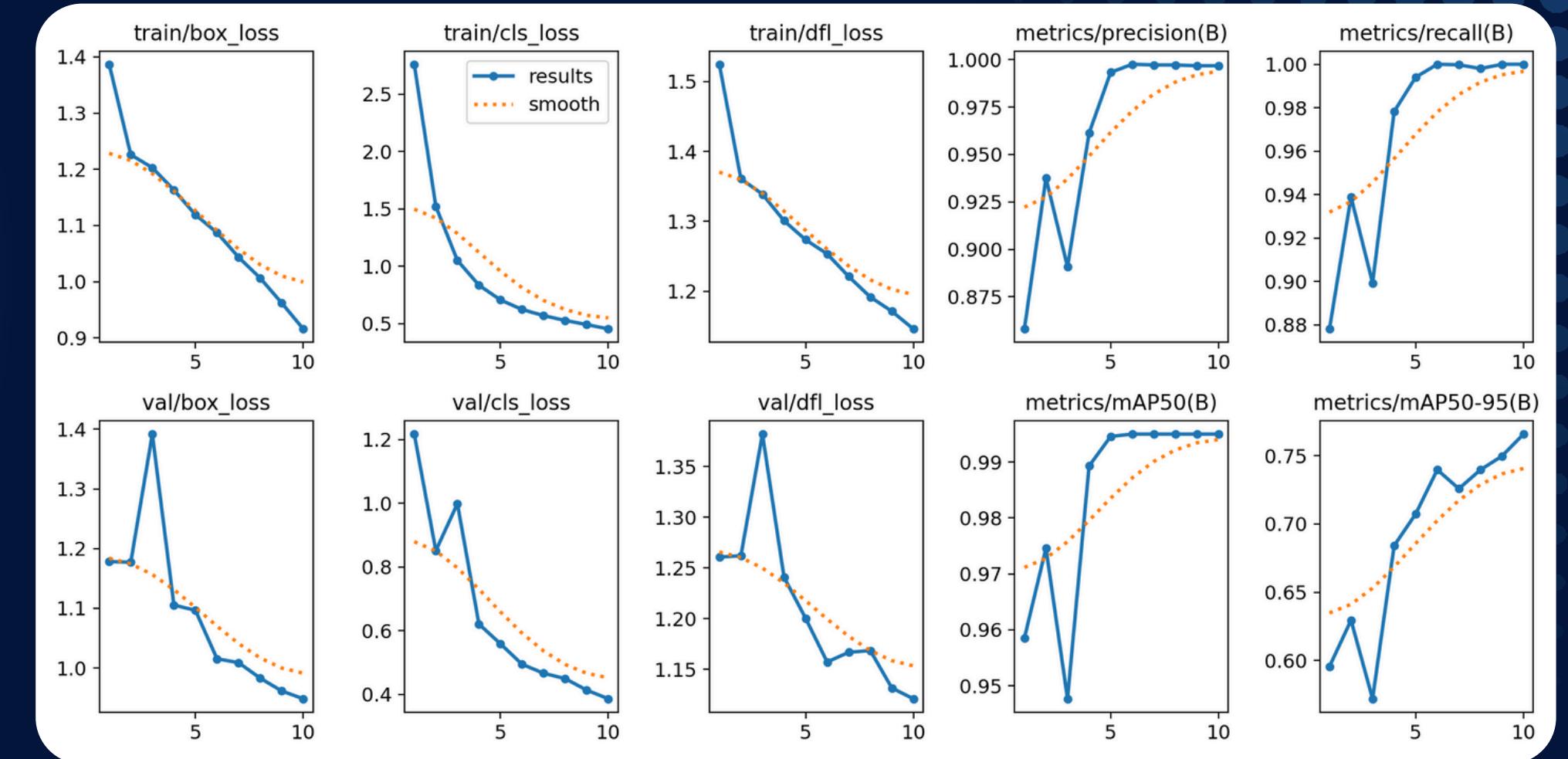
- Lightweight, fast nano version
- Ideal for Raspberry Pi
- Features:
  - CNN backbone
  - PAN/FPN neck
  - Detection head (bounding box + class prediction)

A white, minimalist-style robot head is shown from a three-quarter perspective. It has a large, light blue circular sensor on its forehead with a blue outline and a green curved arrow indicating rotation. A blue magnifying glass is held up to the sensor, focusing on it. The robot's body is white with blue highlights and a small antenna on top.

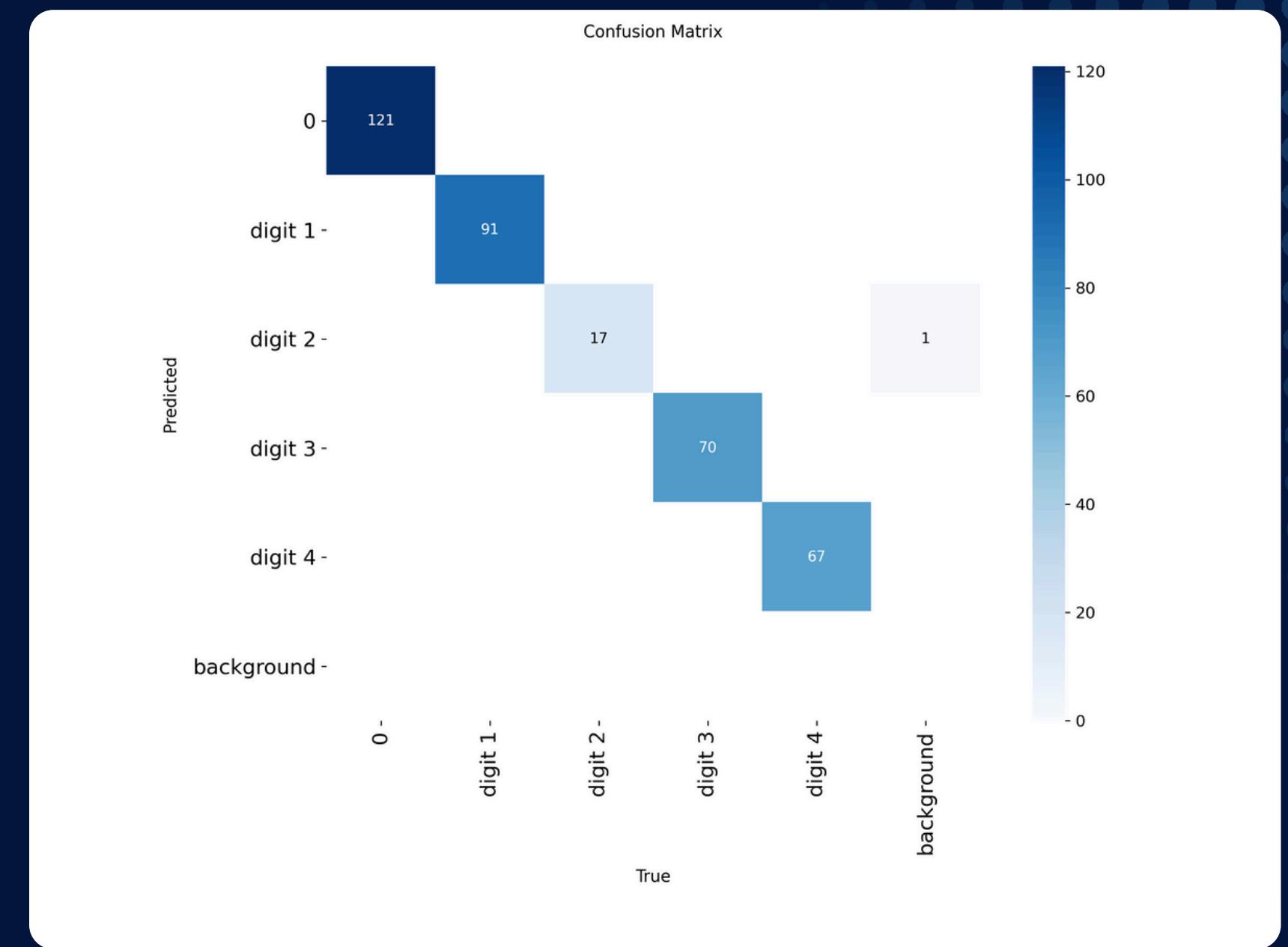
# Training Configuration

- Pretrained: yolov8n.pt
- Epochs (for results): 10
- Image size: 640×640
- Batch size: 16
- Augmentation: rotation, blur, brightness, perspective
- Learned with Roboflow + Ultralytics YOLO

# Training Results



# Confusion Matrix



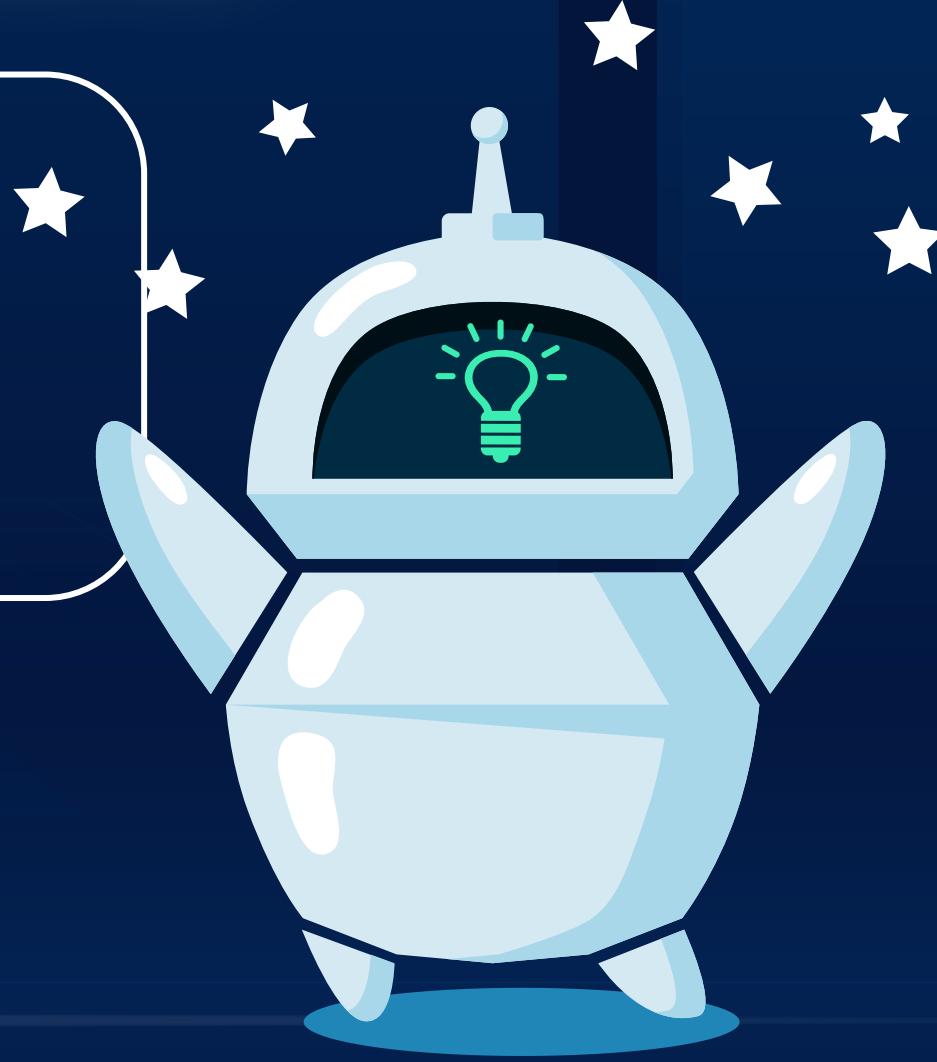
# ONNX Deployment on Raspberry Pi 4

- Model exported to ONNX: best.onnx
- Runs with ONNX Runtime
- Inference is fast enough for live robot control
- Connected to robot movement logic

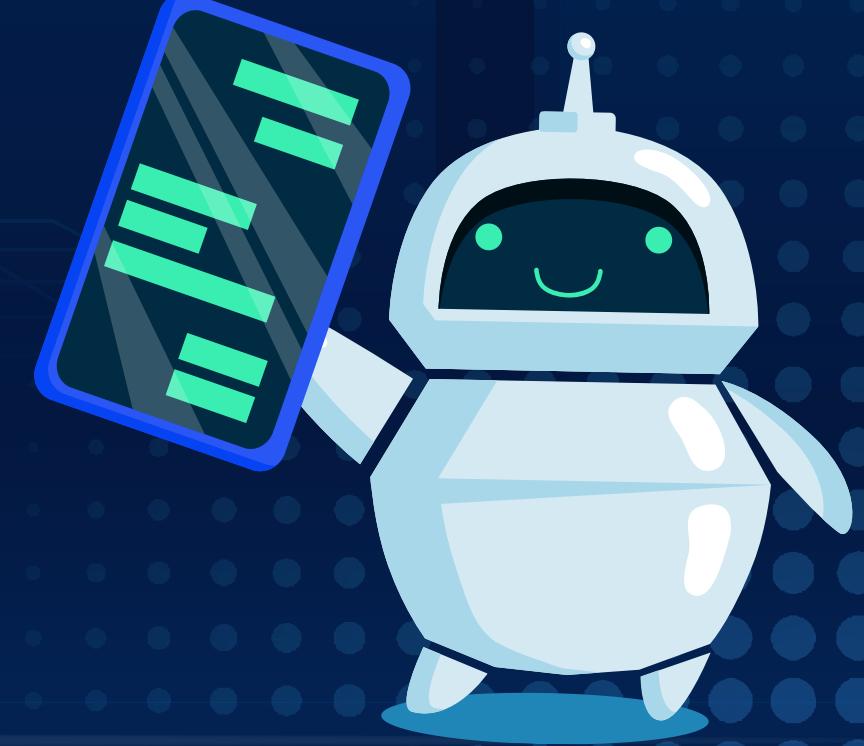
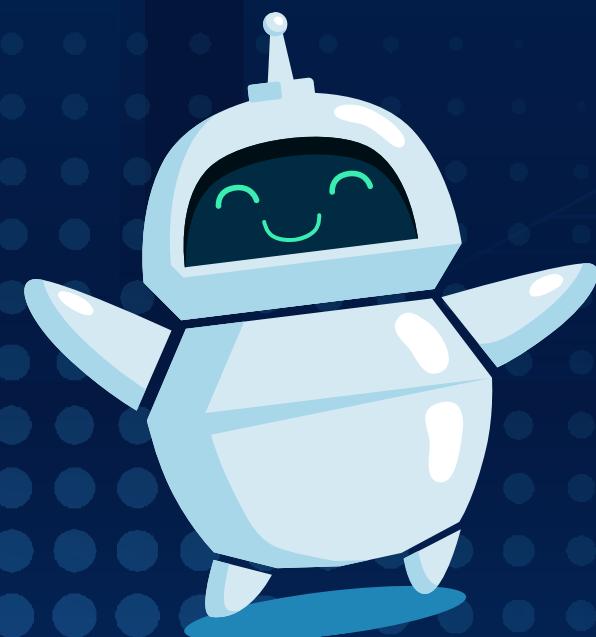


# Challenges & Lessons Learned

- Dataset variation was important for performance
- Lightning problems caused early training issues
- ONNX conversion required correct opset
- Learned how to integrate ML model with robotics



# Demo Video



# Thank You!

