ECE 4380

AI Final Project Presentation

Ring Camera with AI

Team Name: AI Hardware Team
Nate Owen, Marissa Cash, Sammie Levine, Grayson Turner

Overview

- Use the **Arduino Nano 33 BLE Sense Lite with OV7576 Camera Module** to detect specific human faces
- Identify known “Residents” from unknown “Intruders” through the use of facial recognition
- Simulate real world behavior of a Ring Doorbell Camera to associate different home visitors with specific labels

Team Roles

Name	Role	Responsibilities
Grayson Turner	Team Lead	Coordination, documentation
Nate Owen	Hardware	Setup, integration
Sammie Levine	Software	Model training, inference
Marissa Cash	Evaluation	Testing, benchmarking

Project Goals

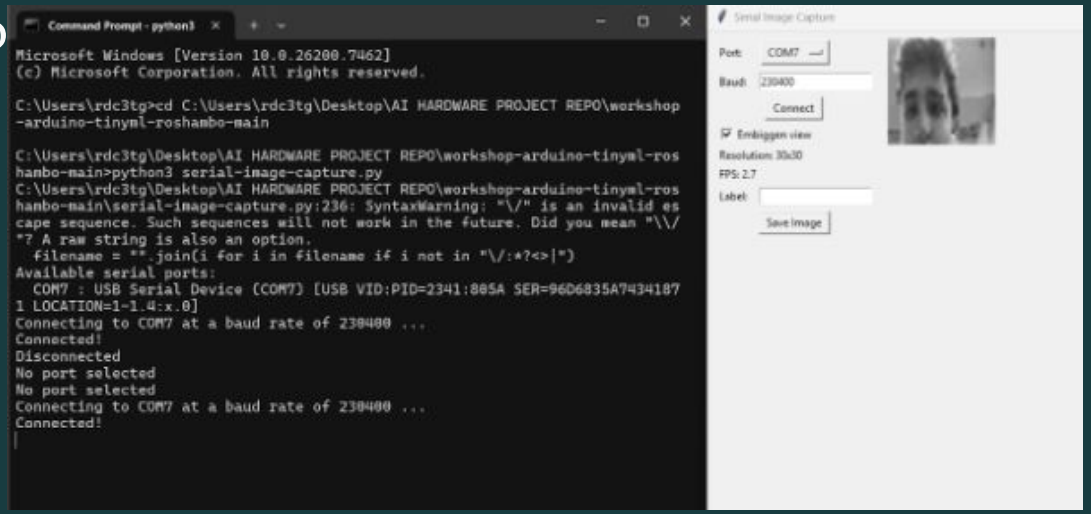
- Train our model to accurately identify unique faces with an above 0.8 success rate in normal lighting conditions
  - Use group members as “known faces” to test facial recognition
- Maintain an average latency of 100ms from image capture to recognition
  - Simulate real world speeds needed to identify visitors at one's door
- Maintain a false positive positive rate of less than 2%
  - Create an accurate model that correctly detects known and unknown visitors

Software

- Built our project using an **Edge Impulse GitHub repository**
  - Repository link: GitHub
  - Adapted the original rock–paper–scissors recognition project to perform facial recognition
- Developed and deployed the software using:
  - **Arduino IDE** for device-side integration
  - **Python** for model handling, data processing, and supporting scripts

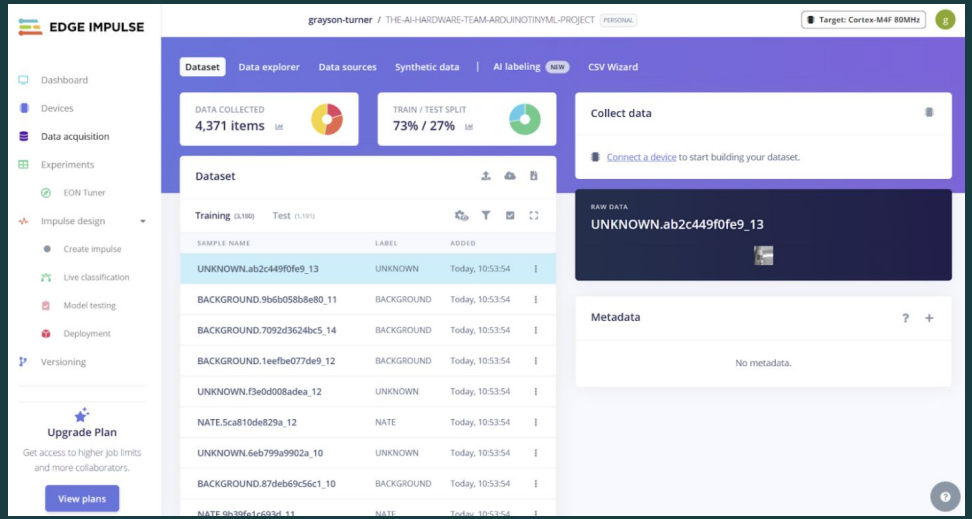
Data Collection

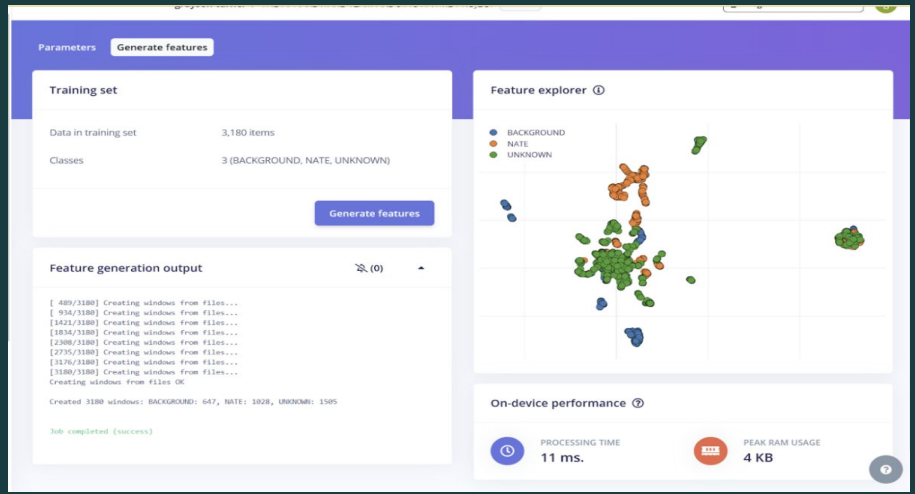
- **Data Collection**:
  - Captured images using the Arduino device camera (OV7675) included in the TinyML kit (Serial Image Capture)
  - Labeled images as:
    - Recognized perso
    - Unknown
    - Background



Data Training

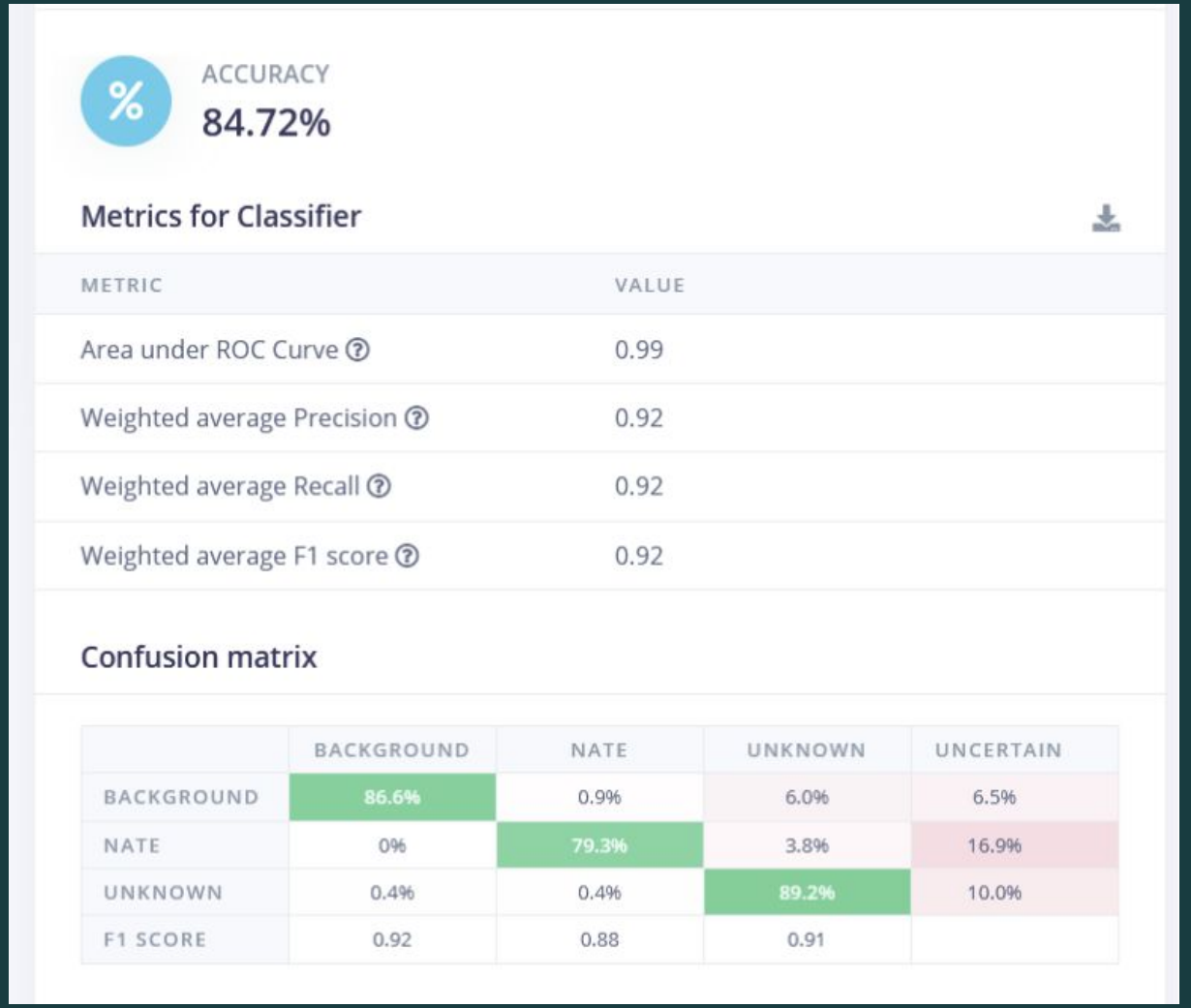
- **Data Training**
  - Create an **Edge Impulse** project and a **Google Colab** project
  - **Google Colab (Data Augmentation):**
    - Generate new training images from each original image
    - Techniques include: flipping, rotation, zooming, translation, and noise addition
  - **Edge Impulse**
    - Train the model and deploy it to run directly on the Arduino device





9

Results



ACCURACY
84.72%

Metrics for Classifier

METRIC	VALUE
Area under ROC Curve	0.99
Weighted average Precision	0.92
Weighted average Recall	0.92
Weighted average F1 score	0.92

Confusion matrix

	BACKGROUND	NATE	UNKNOWN	UNCERTAIN
BACKGROUND	86.6%	0.9%	6.0%	6.5%
NATE	0%	79.3%	3.8%	16.9%
UNKNOWN	0.4%	0.4%	89.2%	10.0%
F1 SCORE	0.92	0.88	0.91	

Results

- Arduino camera successfully identified our known resident (Nate), unknown residents, and background (no face)
- Approximate latency was calculated by adding the inference and DSP latencies together, which are printed in the terminal
  - **RESULT:** The total latency was 141 ms, as the inference latency was reported as 140ms, and the DSP latency was reported as 1ms
- False positive rate was calculated when model identified a face who is not Nate as Nate with >=0.8 certainty out of 10 trials
  - **RESULT:** 1 instance (where the model identified unknown's face as Nate with 0.937 (93.7%) confidence) occurred out of the 10 trials
  - False positive rate occurs at approximately 10%

Future Improvements

Hardware

- Improved camera
  - The OV7576 lacked the resolution and color to accurately differentiate between faces
- More advanced microcontroller
  - Better model training and larger dataset could be used with more computation power

Software

- Larger dataset
  - Add more “known” people to the data