

Face recognition using Portenta H7

Vatsal Mahajan, Manoj Selvaraju, Vijay Singh

January 31, 2025

- 1 Problem's Description
- 2 Description of the Application
- 3 Challenges
- 4 Solutions
- 5 Results
- 6 References

Problem's Description

Problem's Description

Traditional access control systems rely on physical tokens like keys or cards, which can be lost or misused.

- **Objective:** To develop a secure and contactless face recognition system for simple, private and efficient access control using on-device processing.
- **Motivation:** Address the growing demand for efficient authentication by processing data locally on devices, ensuring better privacy and performance.

Description of the Application

Description of the Application

- **DataSet:** Corresponds to three distinct images: Person1, Person2 and Person3.
- **Features:**
 - Facial Feature: Eyes, Nose and Mouth.
 - Image Type: Grayscale images.
- **Models:** A CNN-based MobileNetV2 model that can distinguish between different people by learning important features from their facial images, such as facial expressions, to ensure accurate classification.
- **Hardware:** Arduino Portenta H7, Vision Shield and USB-C cable.
- **Software:** Arduino IDE, Edge Impulse CLI, Python, Nodejs and SolidWorks

Challenges

Challenges

- **Optimised Image resolution:** Building the model using higher resolutions, such as QVGA (320x240) or lower than QQVGA (128x96) from the Vision Shield camera, led to significant overfitting issues.
- **Profile Management:** Attempted to add new profiles and retrain the model in Edge Impulse Studio using API keys, but faced challenges with API integration.
- **Model Accuracy vs. Size:** Keeping the model lightweight for deployment with high accuracy was a challenge.
- **Lightning and Environmental Conditions:** Struggled with varying lightning conditions (e.g., too bright or too dark) or environmental factors (e.g., glare or shadows).
- **Designing the Mounting and Case:** Struggled to give the 3 DOF to the mounted device.

Solutions

Solutions

- **Optimised Image resolution:** Choosing QQVGA (160x120) to balance image quality, reduces overfitting in model building while ensuring accurate recognition.
- **Profile Management:** Preferred manually update the dataset and retrain the model in Edge Impulse Studio [Imp25], prioritizing accuracy over automation.
- **Model Accuracy vs. Size:** Used model Quantization(int8) [Mat25] techniques to reduce the model size without sacrificing accuracy.
- **Lighting and Environmental Conditions:** Ensure optimal hardware placement in environments with balanced lighting to reduce the impact of glare, shadows, and extreme brightness.
- **Designing the Mounting and Case:** Designed the prototype on solidworks.



Abbildung: Portenta Vision Shield Cover

Results

Results I

Configuring Training Parameters

- The optimized model achieved accuracy of **85.2%** during testing, effectively recognized faces across all trained profile.
- Achieved **1789ms** inference time per frame, enabled smooth recognition on Portenta H7.
- Successfully handled varied light conditions with the balanced dataset.
- The system supported profile updates and deletions, ensuring flexible and efficient access control.

Results II

Tabelle: Validation Result

Validation Set	Values
Accuracy	85.2%
Loss	0.41
Average Precision	0.90
Average Recall	0.85
Average F1 Score	0.84

Tabelle: Performance Metrics

Performance Metrics	Values
Inferencing Time	1789ms
Peak Ram Usage	280.9k
Flash Usage	358.2k

Results III

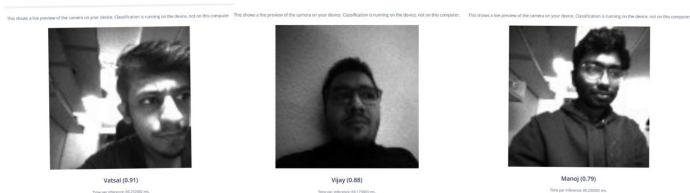


Abbildung: Output of the Face Recognition System

References

References I

- [Ard24a] *Arduino. Portenta H7 Documentation.* 2024. URL: <https://docs.arduino.cc/hardware/portenta-h7/>.
- [Ard24b] *Arduino. Portenta Vision Shield - Arduino Docs.* 2024. URL: <https://docs.arduino.cc/hardware/portenta-vision-shield/#tech-specs>.
- [Gra25a] *GrabCAD. Arduino Portenta H7 Case with the Vision Shield.* 2025. URL: <https://grabcad.com/library/arduino-portenta-h7-case-with-the-vision-shield-1>.
- [Gra25b] *GrabCAD. Raspberry Pi Camera Module Corner Wall Mount.* 2025. URL: <https://grabcad.com/library/raspberry-pi-camera-module-corner-wall-mount-2>.
- [Imp25] *Edge Impulse. API and SDK References.* 2025. URL: <https://docs.edgeimpulse.com/docs/tools/api-and-sdk-references>.

References II

[Mat25] MathWorks. *What is INT8 Quantization and Why is it Popular for Deep Neural Networks?* 2025. URL:
<https://de.mathworks.com/company/technical-articles/what-is-int8-quantization-and-why-is-it-popular-for-deep-neural-networks.html>.

Thank You!!