

# Machine Vision with Arduino Portenta H7

## Face recognition using Edge Impulse and Arduino PortentaH7 connected to Vision Shield.

Vatsal Mahajan, Manoj Selvaraju, Vijay Singh

September 11, 2024

# Table of Content I

- 1 Introduction
- 2 Dataset Preparation and Upload
- 3 Uploading Images
- 4 Impulse Design
- 5 Model Training
- 6 Model Testing
- 7 Model Deployment

# Introduction

Edge Impulse offers a cloud-based system for data collection, neural network design, model training, testing and deployment.

The key steps are:

- Dataset Preparation and Upload.
- Choose the best model for the task.
- Model Training.
- Model Testing and Export.

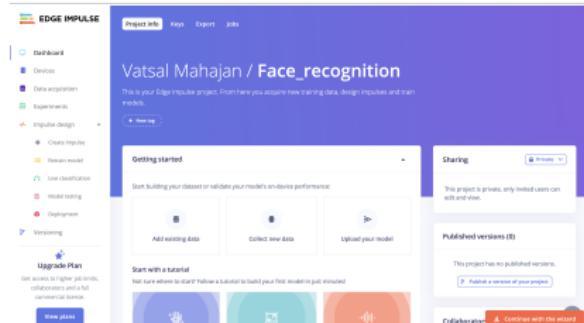


Figure 1: Edge Impulse

# Dataset Preparation and Upload

## Creating Your Own Dataset

In the Project we are using our own dataset for the model training and testing.

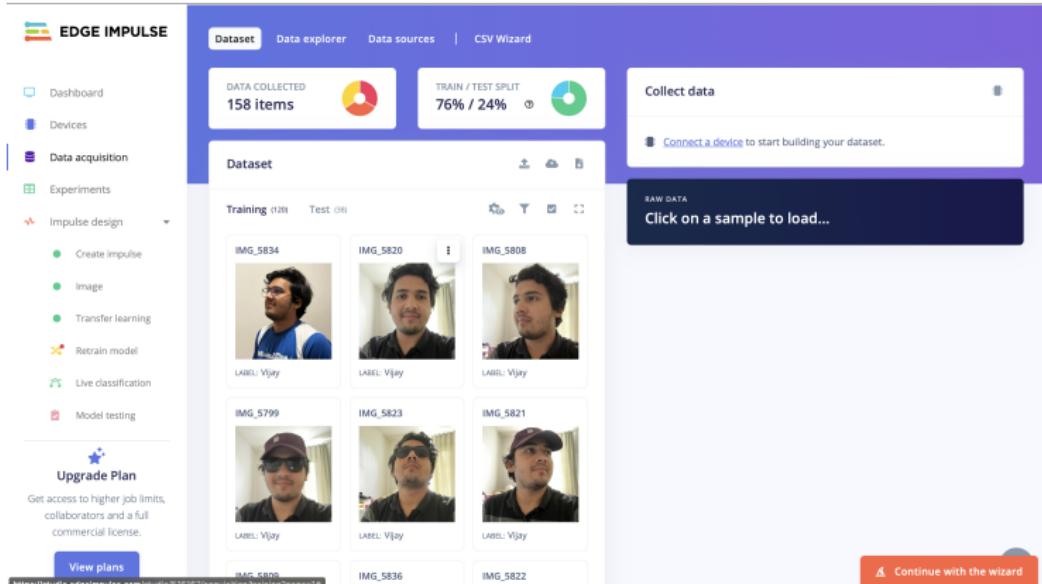
- We have face images for the three distinct people, organize them in separate folders named Vatsal, Manoj, and Vijay.
- To train the model we need enough images (at least 50-70 per person) under different conditions (lighting, angle) for better performance.

## Uploading the dataset to Edge Impulse

- Go to Edge Impulse and log in or create an account.
- Create a new project, name it (e.g., "Face Recognition"), and select Image as the type of data for your project.
- Go to the "Data Acquisition" tab: This is where you'll upload your dataset.

# Uploading Images

- Click on Upload Data. You can drag and drop the images of each person into the platform.
- Make sure to label each image accordingly, like Vatsal, Manoj, and Vijay.



The screenshot shows the Edge Impulse web interface for managing datasets. On the left, a sidebar menu includes options like Dashboard, Devices, Data acquisition, Experiments, Impulse design (with sub-options for Create impulse, Image, Transfer learning, Retrain model), Live classification, Model testing, and an Upgrade Plan section. The main area is titled 'Dataset' and displays a summary: 'DATA COLLECTED 158 items' and 'TRAIN / TEST SPLIT 76% / 24%'. Below this, a 'Collect data' section prompts to 'Connect a device to start building your dataset.' A large dark blue callout box at the bottom right says 'Click on a sample to load...'. The central part of the screen shows a grid of image thumbnails, each with a label below it. The visible labels are: 'IMG\_5834 LABEL: Vijay', 'IMG\_5820 LABEL: Vijay', 'IMG\_5808 LABEL: Vijay', 'IMG\_5799 LABEL: Vijay', 'IMG\_5823 LABEL: Vijay', and 'IMG\_5821 LABEL: Vijay'. At the very bottom, there's a red button labeled 'Continue with the wizard'.

# Impulse Design

Once our dataset is uploaded, we'll design an Impulse (model pipeline) in Edge Impulse.

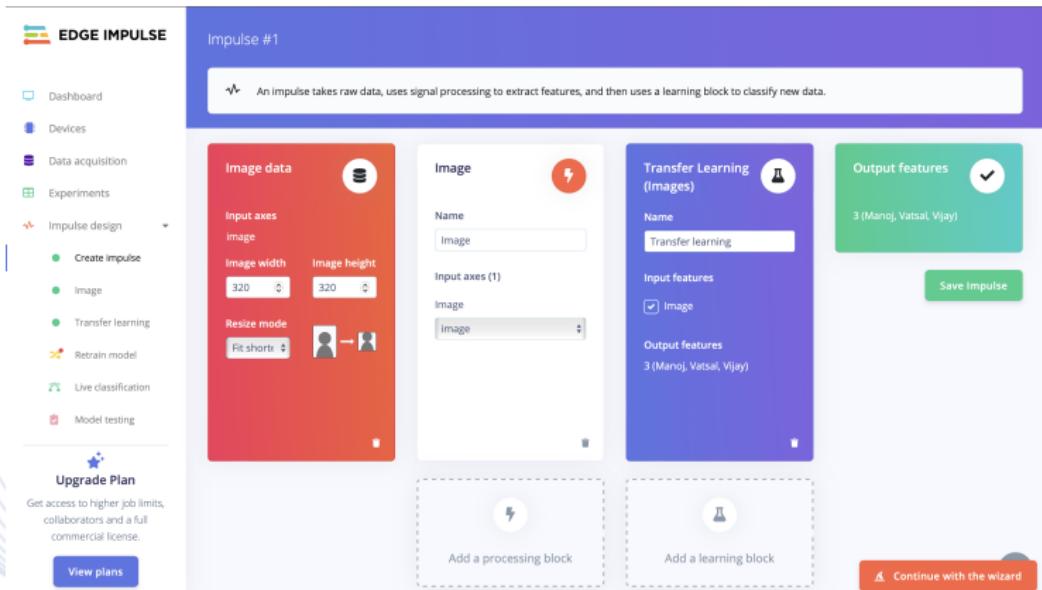
## Choosing the Model Architecture

Model Choice: Convolutional Neural Network (CNN).

For image-related tasks like face recognition, CNNs are typically the best choice. CNNs can detect patterns (like facial features) and differentiate between individuals.

- Click on the Impulse Design tab in Edge Impulse.
- Choose the Image block. This will preprocess the images (resize, convert to grayscale if needed, etc.) and prepare them for model input.
- Select Image Classification as the task.
- Choose a CNN architecture for the project.
- We may need to resize our input images to a  $320 \times 320$  pixels resolution to make the model efficient.

# Impulse Design



The screenshot shows the Edge Impulse web interface for creating a machine learning model. The left sidebar includes links for Dashboard, Devices, Data acquisition, Experiments, Impulse design (with sub-options: Create impulse, Image, Transfer learning, Retrain model, Live classification, Model testing), and an Upgrade Plan section.

The main area is titled "Impulse #1" and contains the following components:

- Image data**: Set to "Image" mode with input axes of 320x320. Resize mode is set to "Fit short".
- Image**: A configuration panel for an "Image" input, showing a preview thumbnail and a red lightning bolt icon.
- Transfer Learning (Images)**: A configuration panel for transfer learning, showing a name "Transfer learning", input features set to "Image", and output features "3 (Manoj, Vatsal, Vijay)".
- Output features**: A green panel showing the output features: "3 (Manoj, Vatsal, Vijay)" with a checked checkbox.

At the bottom, there are dashed boxes for "Add a processing block" and "Add a learning block", and a red "Continue with the wizard" button.

Figure 3: Impulse Design

# Model Training

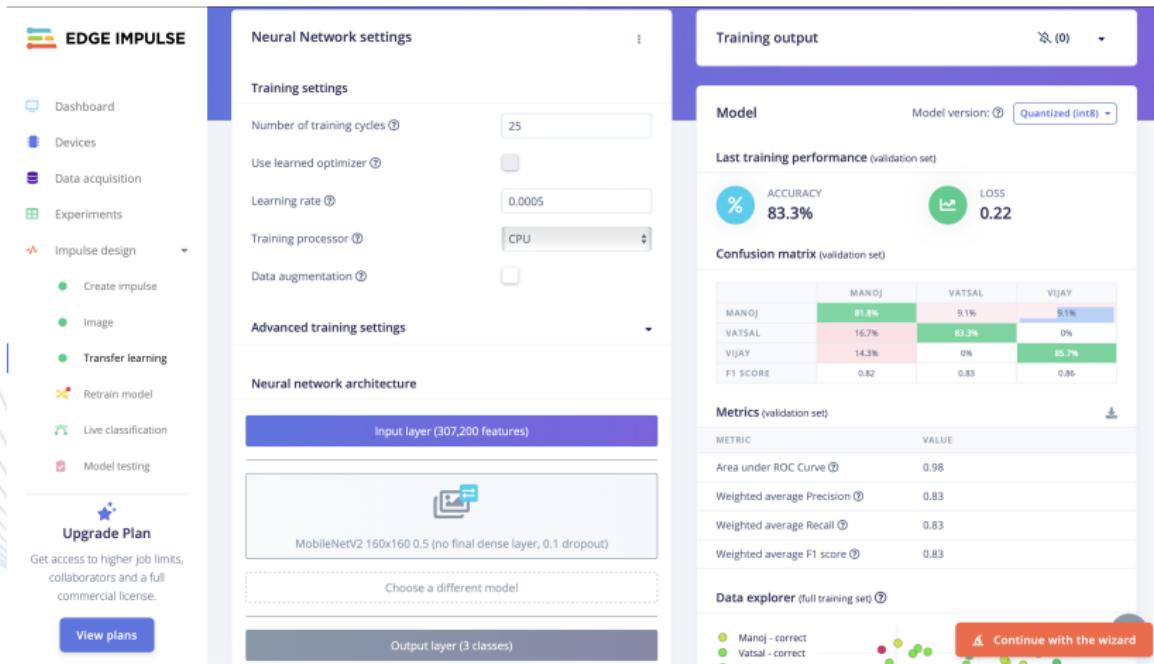
## Configuring Training Parameters

- **Training/Validation Split:** Usually 80% training, 20% validation.  
But in our case, it's **76%** training, **24%** validation.
- **Number of training cycles (Epochs):** 25
- **Learning rate:** 0.0005

## Model Training

- **Accuracy:** 83.3%
- **Loss:** 0.22

# Model Training



The screenshot shows the Edge Impulse web interface for model training. On the left, a sidebar menu includes: Dashboard, Devices, Data acquisition, Experiments, Impulse design (with sub-options Create impulse, Image, Transfer learning, Retrain model), Live classification, Model testing, and an Upgrade Plan section. The Upgrade Plan section encourages users to get access to higher job limits, collaborators, and a full commercial license, with a 'View plans' button.

The main content area is divided into several sections:

- Neural Network settings**:
  - Training settings: Number of training cycles (25), Use learned optimizer (disabled), Learning rate (0.0005), Training processor (CPU), Data augmentation (disabled).
  - Advanced training settings (dropdown menu).
  - Neural network architecture:
    - Input layer (307,200 features)**: MobileNetV2 160x160 0.5 (no final dense layer, 0.1 dropout).
    - Choose a different model (button).
    - Output layer (3 classes)**.
- Training output**: Model version: Quantized (int8). Last training performance (validation set): Accuracy 83.3%, Loss 0.22. Confusion matrix (validation set):

	MANOJ	VATSAL	VIJAY
MANOJ	81.8%	9.1%	9.1%
VATSAL	16.7%	83.3%	0%
VIJAY	14.3%	0%	85.7%
F1 SCORE	0.82	0.83	0.86

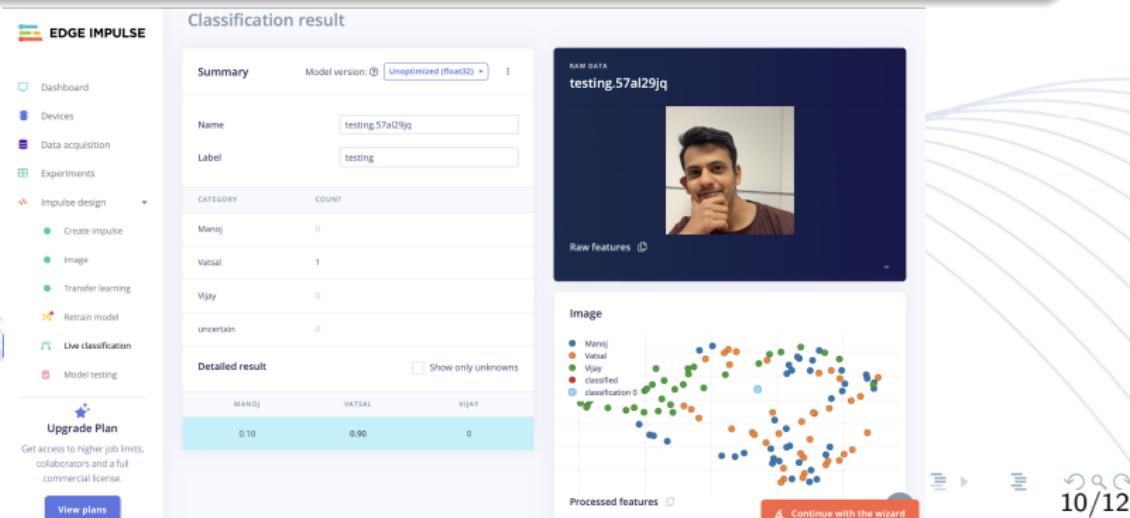
Metrics (validation set): Area under ROC Curve 0.98, Weighted average Precision 0.83, Weighted average Recall 0.83, Weighted average F1 score 0.83. Data explorer (full training set) shows a scatter plot of data points for Manoj - correct and Vatsal - correct categories.

Figure 4: Model Training Progress

# Model Testing

## Evaluating the Model

- **To Test the model:** Go to "Live Classification" tab. we uploaded some new images and the model recognizes faces.
- **Confidence Score:** The system will show confidence scores for each class (Vatsal, Manoj, Vijay).



The screenshot shows the Edge Impulse web interface. On the left is a sidebar with navigation links: Dashboard, Devices, Data acquisition, Experiments, Impulse design (with sub-options Create impulse, Image, Transfer learning, Retrain model), Live classification, Model testing, and Upgrade Plan. The main area is titled "Classification result". It displays a "Summary" section with "Model version: Unoptimized (float32)" and a table of results:

Name	Label
testing.57a129jq	testing
Manoj	0
Vatsal	1
Vijay	0
uncertain	0

Below this is a "Detailed result" section with a "Show only unknowns" checkbox. A table shows confidence scores for three classes:

	MANOJ	VATSAL	VIJAY
	0.10	0.90	0

On the right, there is a "RAW DATA" section showing a portrait of a man and a "Raw features" section. At the bottom is a "Processed features" scatter plot with points colored by class: Manoj (blue), Vatsal (orange), Vijay (green), and Unclassified (grey). A legend indicates "classification D".

# Model Deployment

## Evaluating the Model

- Go to the Deployment tab.
- Select **TensorFlow Lite (.tflite)** as the export format.
- **Quantization:** Choose **int8 Quantization** if you plan to run the model on embedded systems like Portenta H7. Quantization reduces the model size and speeds up inference.

Thank you  
for your attention