Edge Computing Lab

Name : Rahul Bhati

Class: TY-AIEC

Roll no : 2223416

School of Computing, MIT Art Design Technology University

Academic Year: 2024-25

Experiment No. 6

Title

Keyword Spotting Project like “OK, Google,” “Alexa,” on Edge Devices using Microphone

Objective: Build a project to detect the keywords using a built-in sensor on Nano BLE Sense / Mobile Phone

Tasks:

- Generate the dataset for keyword
- Configure BLE Sense / Mobile for Edge Impulse
- Building and Training a Model

Run the project Keyword Spotting like “OK, Google,” “Alexa

Introduction

Edge Impulse is a development platform for machine learning on edge devices, targeted at developers who want to create intelligent device solutions. The "Hello World" equivalent in Edge Impulse would typically involve creating a simple machine learning model that can run on an edge device, like classifying sensor data or recognizing a basic pattern.

Materials Required

- Nano BLE Sense Board

Theory

GPIO (General Purpose Input/Output) pins on the Raspberry Pi are used for interfacing with other electronic components. BCM numbering refers to the pin numbers in the Broadcom SOC channel, which is a more consistent way to refer to the GPIO pins across different versions of the

Here’s a high-level overview of steps you'd follow to create a "Hello World" project on Edge Impulse:

Steps to Configure the Edge Impulse:

1. Create an Account and New Project:
    - Sign up for an Edge Impulse account.
    - Create a new project from the dashboard.

2. Connect a Device:

   - You can use a supported development board or your smartphone as a sensor device.
   - Follow the instructions to connect your device to your Edge Impulse project.

3. Collect Data:

   - Use the Edge Impulse mobile app or the Web interface to collect data from the onboard sensors.
   - For a "Hello World" project, you could collect accelerometer data, for instance.

4. Create an Impulse:

   - Go to the 'Create impulse' page.
   - Add a processing block (e.g., time-series data) and a learning block (e.g., classification).
   - Save the impulse, which defines the machine learning pipeline.

5. Design a Neural Network:

   - Navigate to the 'NN Classifier' under the 'Learning blocks'.
   - Design a simple neural network. Edge Impulse provides a default architecture that works well for most basic tasks.

6. Train the Model:

   - Click on the 'Start training' button to train your machine learning model with the collected data.

7. Test the Model:

   - Once the model is trained, you can test its performance with new data in the 'Model Testing' tab.

8. Deploy the Model:

   - Go to the 'Deployment' tab.
   - Select the deployment method that suits your edge device (e.g., Arduino library, WebAssembly, container, etc.).

- Follow the instructions to deploy the model to your device.

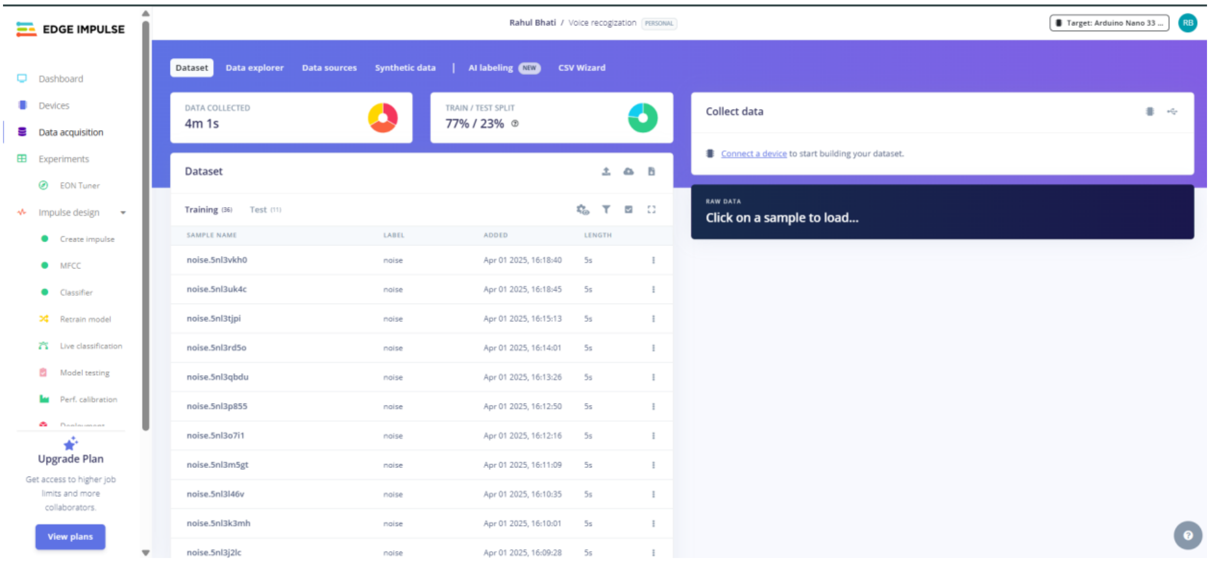
9. Run Inference:

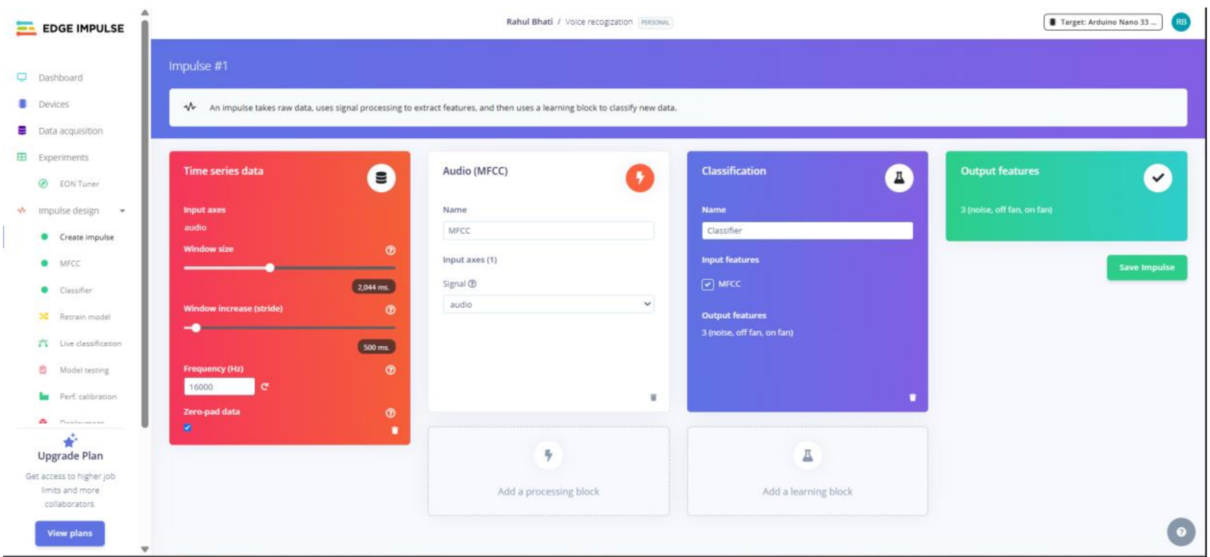
- With the model deployed, run inference on the edge device to see it classifying data in real-time.

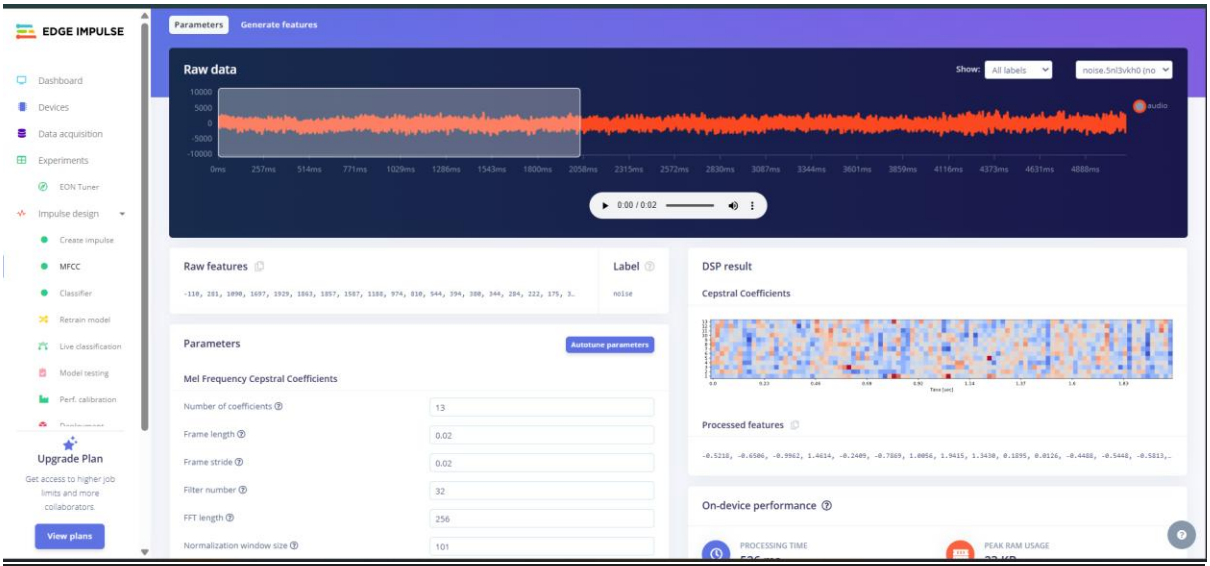
10. Monitor:

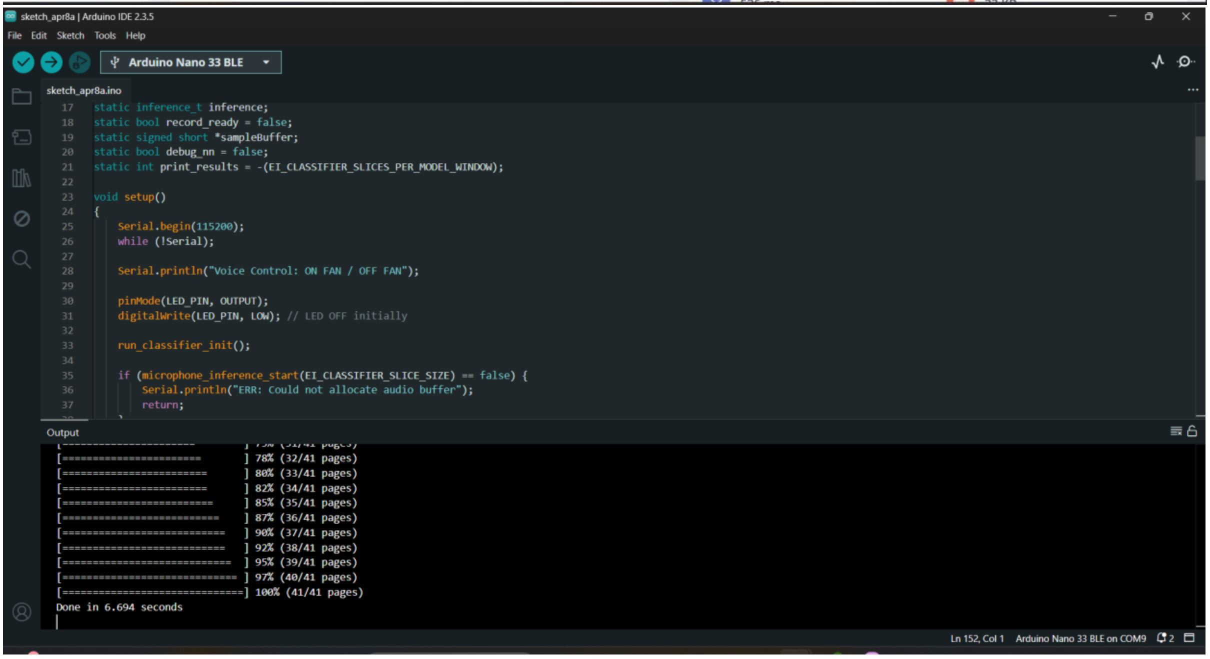
- You can monitor the performance of your device through the Edge Impulse studio.

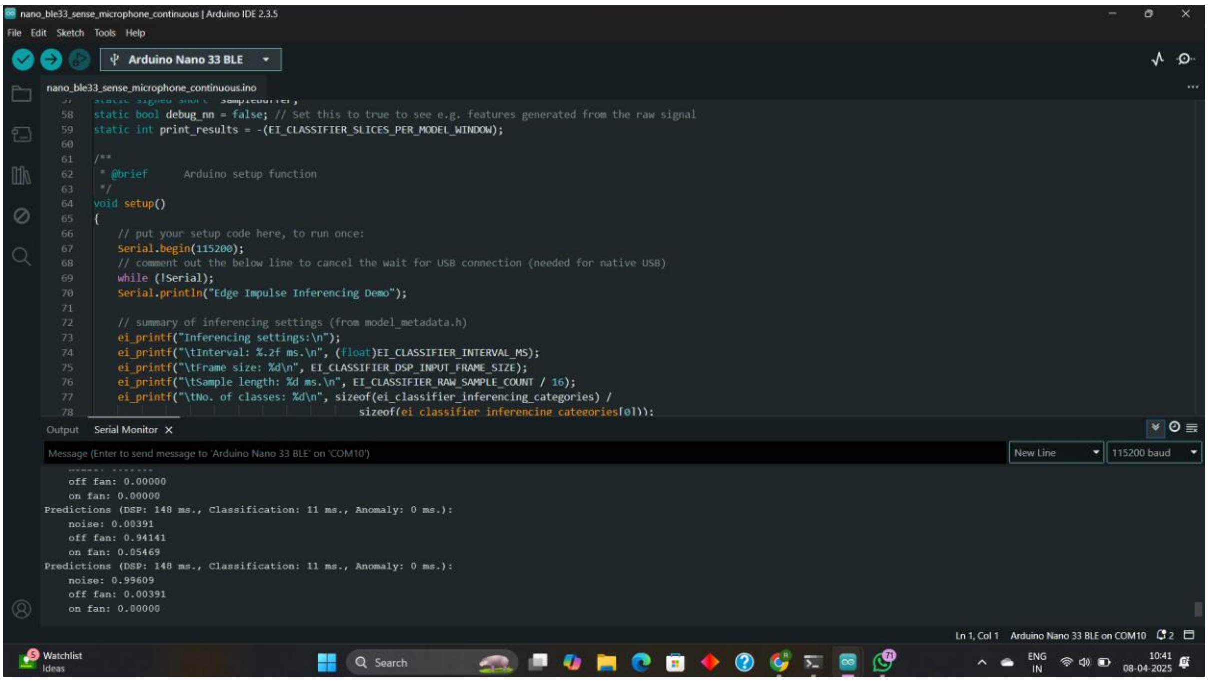
Edge Impulse project's Results:






EDGE IMPULSE
Dashboard
Devices
Data acquisition
Experiments
EON Tuner
Impulse design
Create impulse
MFCC
Classifier
Retrain model
Live classification
Model testing
Perf. calibration
Upgrade Plan
Get access to higher job limits and more collaborators
View plans
Parameters
Generate features
Raw data
Show:
All labels
Raw features
Label
noise
DSP result
Cepstral Coefficients
Parameters
Autotune parameters
Mel Frequency Cepstral Coefficients
Number of coefficients
13
Frame length
0.02
Frame stride
0.02
Filter number
32
FFT length
256
Normalization window size
101
Processed features
On-device performance
PROCESSING TIME
PEAK RAM USAGE


sketch_apr8a | Arduino IDE 2.3.5
File Edit Sketch Tools Help
Arduino Nano 33 BLE
sketch_apr8a.ino
static inference_t inference;
static bool record_ready = false;
static signed short *sampleBuffer;
static bool debug_nn = false;
static int print_results = -(EI_CLASSIFIER_SLICES_PER_MODEL_WINDOW);
void setup()
{
Serial.begin(115200);
while (!Serial);
Serial.println("Voice Control: ON FAN / OFF FAN");
pinMode(LED_PIN, OUTPUT);
digitalWrite(LED_PIN, LOW); // LED OFF initially
run_classifier_init();
if (microphone_inference_start(EI_CLASSIFIER_SLICE_SIZE) == false) {
Serial.println("ERR: Could not allocate audio buffer");
return;
Output
] 78% (32/41 pages)
] 80% (33/41 pages)
] 82% (34/41 pages)
] 85% (35/41 pages)
] 87% (36/41 pages)
] 90% (37/41 pages)
] 92% (38/41 pages)
] 95% (39/41 pages)
] 97% (40/41 pages)
] 100% (41/41 pages)
Done in 6.694 seconds
Ln 152, Col 1 Arduino Nano 33 BLE on COM9


nano_ble33_sense_microphone_continuous | Arduino IDE 2.3.5
File Edit Sketch Tools Help
Arduino Nano 33 BLE
nano_ble33_sense_microphone_continuous.ino
static bool debug_nn = false; // Set this to true to see e.g. features generated from the raw signal
static int print_results = -(EI_CLASSIFIER_SLICES_PER_MODEL_WINDOW);
/**
* @brief Arduino setup function
*/
void setup()
{
// put your setup code here, to run once:
Serial.begin(115200);
// comment out the below line to cancel the wait for USB connection (needed for native USB)
while (!Serial);
Serial.println("Edge Impulse Inferencing Demo");
// summary of inferencing settings (from model_metadata.h)
ei_printf("Inferencing settings:\n");
ei_printf("\tInterval: %.2f ms.\n", (float)EI_CLASSIFIER_INTERVAL_MS);
ei_printf("\tFrame size: %d\n", EI_CLASSIFIER_DSP_INPUT_FRAME_SIZE);
ei_printf("\tSample length: %d ms.\n", EI_CLASSIFIER_RAW_SAMPLE_COUNT / 16);
ei_printf("\tNo. of classes: %d\n", sizeof(ei_classifier_inferencing_categories) /
sizeof(ei_classifier_inferencing_categories[0]));
Output
Serial Monitor
Message (Enter to send message to 'Arduino Nano 33 BLE' on 'COM10')
New Line
115200 baud
off fan: 0.00000
on fan: 0.00000
Predictions (DSP: 148 ms., Classification: 11 ms., Anomaly: 0 ms.):
noise: 0.00391
off fan: 0.94141
on fan: 0.05469
Predictions (DSP: 148 ms., Classification: 11 ms., Anomaly: 0 ms.):
noise: 0.99609
off fan: 0.00391
on fan: 0.00000
Ln 1, Col 1
Arduino Nano 33 BLE on COM10
Watchlist
Ideas
Search
ENG
IN
10:41
08-04-2025