

Universidad de Costa Rica
Facultad de Ingeniería
Escuela de Ingeniería Eléctrica

IE-0624 Laboratorio de Microcontroladores

Arduino Nano 33 BLE: GPIO y TinyML

Wilber Hernández Ruiz B13257

8 de julio del 2024

1. Introducción

Este laboratorio trata del uso de la herramienta EdgeImpulse para hacer un modelo de *machine learning* a partir de un conjunto de datos. El modelo de aprendizaje es utilizado para que se pueda detectar por medio de una cámara si una persona esta utilizando o uno una mascarilla. El modelo es compilado para que sea utilizado en el microcontrolador Arduino Nano BLE 33 Sense Lite, perteneciente al kit de machine learning de Arduino.



Figura 1: Kit de machine learning utilizado para el laboratorio.

2. Nota Teórica

El Arduino Nano BLE 33 Sense Lite TinyML kit posee las siguientes características:

- Es una placa de desarrollo compacta y versatil con 11 sensores.
- Placa diseñada para aplicaciones de bajo consumo y tiene conectividad BLE (Bluetooth Low Energy)
- IMU LSM9DS1 de 9-ejes (Giroscopio, acelerometro, magnetometro)
- Sensor de proximidad, color RGB, intensidad de luz y detección de gestos APDS9960
- Micrófono digital, cámara OV7675
- Versión Lite no tiene el sensor HTS221 (Temperatura y humedad) pero tiene el LPS22HB sensor de presión y temperatura

■ Soporte de MicroPython

De las características antes mencionadas, se va utilizar la cámara OV7675 para captar las imágenes que van a formar parte de la base de datos, en una plataforma de *machine learning* llamada Edge Impulse. Dicha plataforma tiene a disposición los procedimientos necesarios para realizar el diseño y entrenamiento de modelos a partir de conjunto de datos ingresado por el usuario. Los modelos predeterminados son el reconocimiento de patrones para audio, imágenes y movimiento. Presenta un bloque de preprocesamiento de los datos para la generación de los vectores de características. Para realizar el entrenamiento del modelo tiene a disposición el uso de redes neuronales.

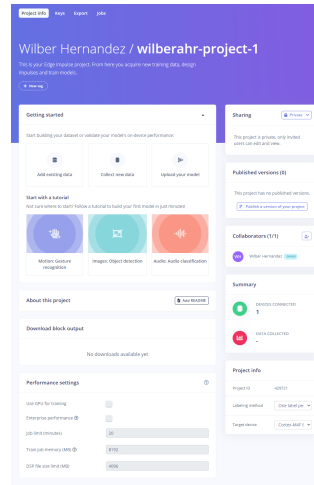


Figura 2: Vista general de proyecto en Edge Impulse.

3. Metodología

Se realizó un entrenamiento supervisado de un modelo de reconocimiento de imágenes, utilizando etiquetado de imágenes para "cara" (face) y "máscara" (mask). El entrenamiento es supervisado en tanto se utiliza una razón 80/20 en donde el 80 % del conjunto de datos es para el entrenamiento del modelo y el restante 20 % es utilizado en la prueba del modelo. Para ello se tomaron alrededor de 60 imágenes, 20 con mascara, 20 sin mascara, y 20 solo con el fondo.

Se realizó la instalación de varios paquetes de software. Primero se instaló el Arduino CLI y Node.js. En la figura 3 se muestra la instalación de Arduino CLI. Luego se hizo la instalación de Edge Impulse en la figura 4 y se ejecutó para realizar la conexión entre la plataforma web de Edge Impulse y el microcontrolador en la figura 5, el cual fue cargado con el firmware que permitiera acceder al feed de la cámara desde Edge Impulse y así hacer la toma de datos.

```
C:\WINDOWS\System32\cmd.exe

You're using an untested version of Arduino CLI, this might cause issues (found: 1.0.2, expected: 0.33.x)
You're using an untested version of Arduino CLI, this might cause issues (found: 1.0.2, expected: 0.33.x)
Finding Arduino Mbed core...
arduino:mbed nano 4.1.3 4.1.3 Arduino Mbed OS Nano Boards
Finding Arduino Mbed core OK
Finding Arduino Nano 33 BLE...
Finding Arduino Nano 33 BLE OK at COM8
arduino:mbed nano 4.1.3 4.1.3 Arduino Mbed OS Nano Boards
Device : nRF52840-QTAA
Version : Arduino Bootloader (SAM-BA extended) 2.0 [Arduino:IKXYZ]
Address : 0x0
Pages : 256
Page Size : 4096 bytes
Total Size : 1024KB
Planes : 1
Lock Regions : 0
Locked : none
Security : false
Erase flash

Done in 0.001 seconds
Write 303504 bytes to flash (75 pages)
[=====] 100% (75/75 pages)
Done in 12.316 seconds
New upload port: COM8 (serial)
Flashed your Arduino Nano 33 BLE development board
To set up your development with Edge Impulse, run 'edge-impulse-daemon'
To run your impulse on your development board, run 'edge-impulse-run-impulse'
Press any key to continue . . .
```

Figura 3: Instalación de Arduino CLI.

```
C:\Users\wilber\Downloads\install-scripts-main\install-scripts-main>npm install -g edge-impulse-cli@1.26.0 --force
npm warn using --force Recommended protections disabled.
npm warn deprecated osmpt@1.0: This package is no longer supported.
npm warn deprecated inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out lru-cache if you want a good and tested way to coalesce async requests by a key value, which is much more comprehensive and powerful.
npm warn deprecated @zeit/dockerignore@0.5.1: '@zeit/dockerignore' is no longer maintained
npm warn deprecated rimraf@2.7.1: Rimraf versions prior to v4 are no longer supported
npm warn deprecated node-ipc@1.2.0: This package is no longer supported.
npm warn deprecated figgy-pudding@3.2.1: This module is no longer supported.
npm warn deprecated normalize-git-url@3.1.1: This library is no longer supported.
npm warn deprecated copy-concurrently@1.0.5: This package is no longer supported.
npm warn deprecated request-promise@4.2.4: request-promise has been deprecated because it extends the now deprecated request package, see https://github.com/request/request/issues/3142
npm warn deprecated glob@7.2.3: Glob versions prior to 8 are no longer supported
npm warn deprecated @aws-sdk/credential-providers@1.7: This package is no longer supported.
npm warn deprecated debug@1.1.1: Debug versions >=1.2.8 <3.2.7 || >=4 <4.3.1 have a low-severity DoS regression when used in a Node.js environment. It is recommended you upgrade to 3.2.7 or 4.3.1. (https://github.com/visionmedia/debug/issues/797)
npm warn deprecated debug@1.1.1: Debug versions >=1.2.8 <3.2.7 || >=4 <4.3.1 have a low-severity DoS regression when used in a Node.js environment. It is recommended you upgrade to 3.2.7 or 4.3.1. (https://github.com/visionmedia/debug/issues/797)
npm warn deprecated debug@1.1.1: Debug versions >=1.2.8 <3.2.7 || >=4 <4.3.1 have a low-severity DoS regression when used in a Node.js environment. It is recommended you upgrade to 3.2.7 or 4.3.1. (https://github.com/visionmedia/debug/issues/797)
npm warn deprecated gauge@2.7.4: This package is no longer supported.
npm warn deprecated @babel/plugin-transform-strict-mode@7.0: This package is no longer supported.
npm warn deprecated @babel/plugin-transform-strict-mode@7.0: This package is no longer supported.
npm warn deprecated request@2.88.0: request has been deprecated, see https://github.com/request/request/issues/3142
npm warn deprecated @babel/plugin-transform-strict-mode@7.0: This package is no longer supported.
added 3 packages, removed 2 packages, and changed 373 packages in 33s
35 packages are looking for funding
  run `npm fund` for details
C:\Users\wilber\Downloads\install-scripts-main\install-scripts-main>edge-impulse-daemon
Edge Impulse serial daemon v1.26.0
Endpoints:
  Websocket: wss://remote-mgmt.edgeimpulse.com
  API: https://studio.edgeimpulse.com
  Ingestion: https://ingestion.edgeimpulse.com
? Which device do you want to connect to? COM8 (Microsoft)
[SER] Connecting to COM8
[SER] Serial is connected, trying to read config...
[SER] Retrieved configuration
[SER] Device is running AI command version 1.8.0
Setting upload host in device... OK
Configuring remote management settings... OK
Configuring API key in device... OK
[SER] Device is not connected to remote management API, will use daemon
[WS] Connecting to wss://remote-mgmt.edgeimpulse.com
[WS] Connected to wss://remote-mgmt.edgeimpulse.com
? What name do you want to give this device? mask-recognition
[WS] Device "mask-recognition" is now connected to project "wilberahh-project-1". To connect to another project, run 'edge-impulse-daemon --clean'.
[WS] Go to https://studio.edgeimpulse.com/studio/429721/acquisition/training to build your machine learning model!
```

Figura 4: Instalación de Edge Impulse.

```
C:\Windows\System32>edge-impulse-daemon
Edge Impulse serial daemon v1.26.0
Endpoints:
  Websocket: wss://remote-mgmt.edgeimpulse.com
  API: https://studio.edgeimpulse.com
  Ingestion: https://ingestion.edgeimpulse.com
? Which device do you want to connect to? COM8 (Microsoft)
[SER] Connecting to COM8
[SER] Serial is connected, trying to read config...
[SER] Retrieved configuration
[SER] Device is running AI command version 1.8.0
Setting upload host in device... OK
Configuring remote management settings... OK
Configuring API key in device... OK
[SER] Device is not connected to remote management API, will use daemon
[WS] Connecting to wss://remote-mgmt.edgeimpulse.com
[WS] Connected to wss://remote-mgmt.edgeimpulse.com
? What name do you want to give this device? mask-recognition
[WS] Device "mask-recognition" is now connected to project "wilberahh-project-1". To connect to another project, run 'edge-impulse-daemon --clean'.
[WS] Go to https://studio.edgeimpulse.com/studio/429721/acquisition/training to build your machine learning model!
```

Figura 5: Configuración de la conexión entre el servidor de Edge Impulse y el Arduino Nano 33 BLE.

4. Análisis de resultados

En la figura 6 se muestra el conjunto de datos obtenido, en este paso para la detección de la máscara. A su vez después de obtener el conjunto de datos se realizó el etiquetado de las imágenes mostrado en la figura 7 para luego hacer la generación del vector de características en la figura 8. Se realizó el entrenamiento del modelo con el 80 % del conjunto de datos, el cual fue escogido aleatoriamente por la plataforma de Edge Impulse.

Dataset

Data sources

Labeling queue (15)

DATA COLLECTED

15 items

TRAIN / TEST S...

-

Dataset

Training (15)

Test (0)

SAMPLE NAME	LABELS	ADDED
with-mask.522mi61o	-	Today, 01:3...
with-mask.522mgtrt	-	Today, 01:3...
with-mask.522mgeei	-	Today, 01:3...
with-mask.522mdqvk	-	Today, 01:3...
with-mask.522mcion	-	Today, 01:3...
with-mask.522mc073	-	Today, 01:3...
with-mask.522m8dq1	-	Today, 01:3...
with-mask.522m84h4	-	Today, 01:3...
with-mask.522m7r8l	-	Today, 01:3...
with-mask.522m7hun	-	Today, 01:3...
with-mask.522m72fe	-	Today, 01:2...
with-mask.522m6j14	-	Today, 01:2...

Collect data

Device

mask-recognition

Name

with-mask

Sensor

Camera (160x120)

Camera feed

Start sampling

RAW DATA

with-mask.522mi61o

Metadata

No metadata.

Figura 6: Toma de imagenes para el set de datos.

Labeling 'with-mask.522m7hun' (15 items left)

Label suggestions: Track objects between frames

Use your mouse to drag a box around an object to add a label. Then click **Save labels** to advance to the next item.

Delete sample

Save labels

Figura 7: Etiquetado de imágenes.

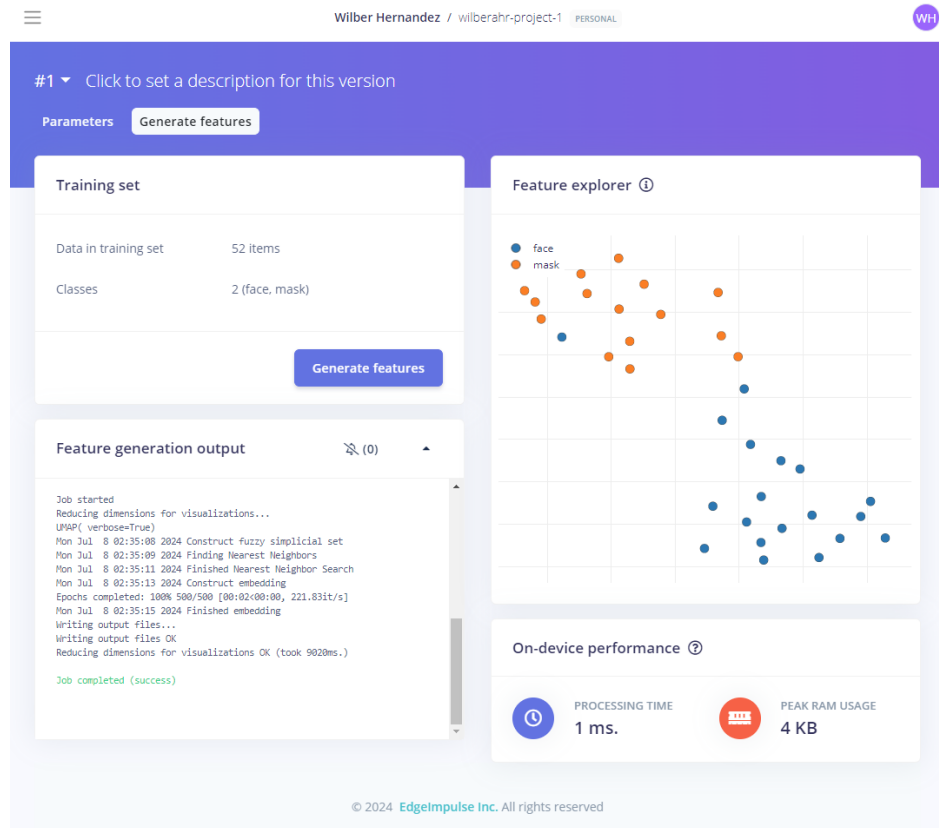


Figura 8: Set de entrenamiento para el proyecto.

Se realiza el entrenamiento del modelo mediante las redes neuronales, mostrado en la figura 9. Se obtiene un rendimiento de 58% de exactitud. Se muestra en la matriz de confusión que el problema radica en que no se logró detectar cara a partir del modelo. Dos posibles razones son que se usó una mascarilla negra y la persona contaba con barba, y eso resultó en que la exactitud estuviera debajo del 70%. La otra razón es el conjunto de datos pequeño que es de solo 60 imágenes. Al no contar con otra mascarilla y persona para generar un mayor conjunto de datos se deja así.

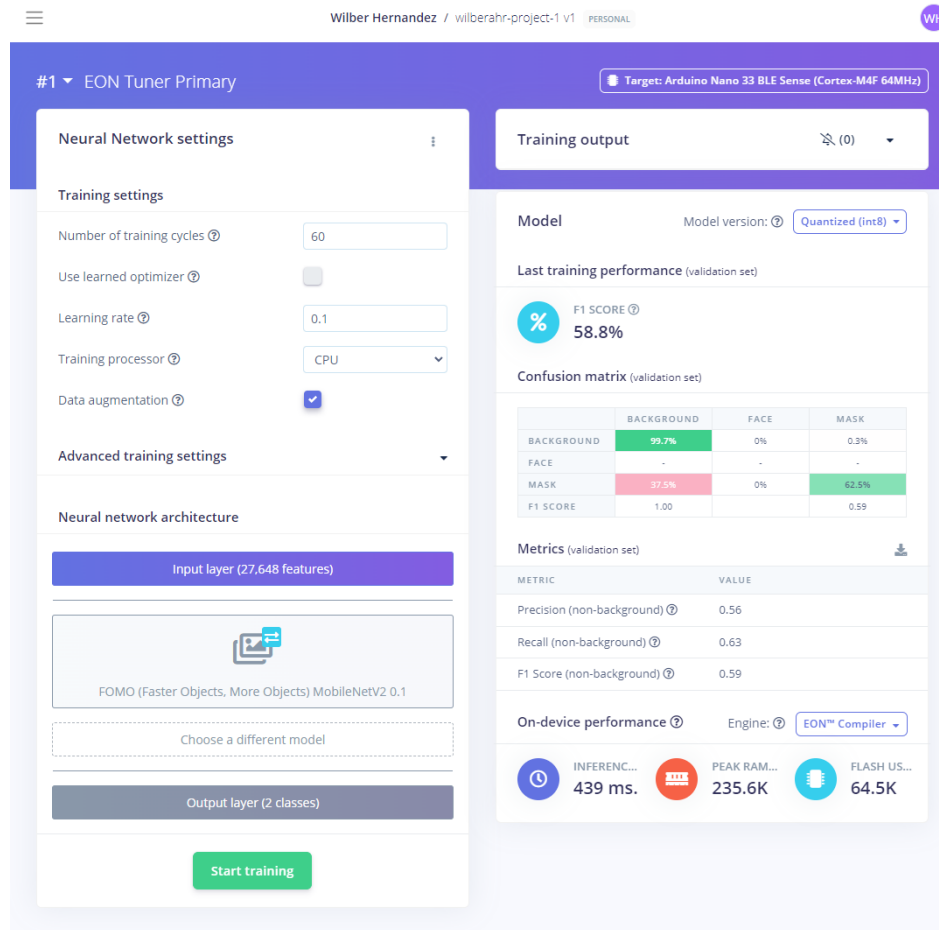


Figura 9: Entrenamiento mediante redes neuronales con Eonturner.

En la figura 10 se realiza la búsqueda del arquitectura para la aplicación mediante EONturner. Se selecciona la 245360 por tener los mejores parámetros de rendimiento y dentro de los límites de ROM y RAM del Arduino Nano BLE 33. Presenta el uso de RGB en lugar de Grayscale. Las demás características de muestran en la figura 11.

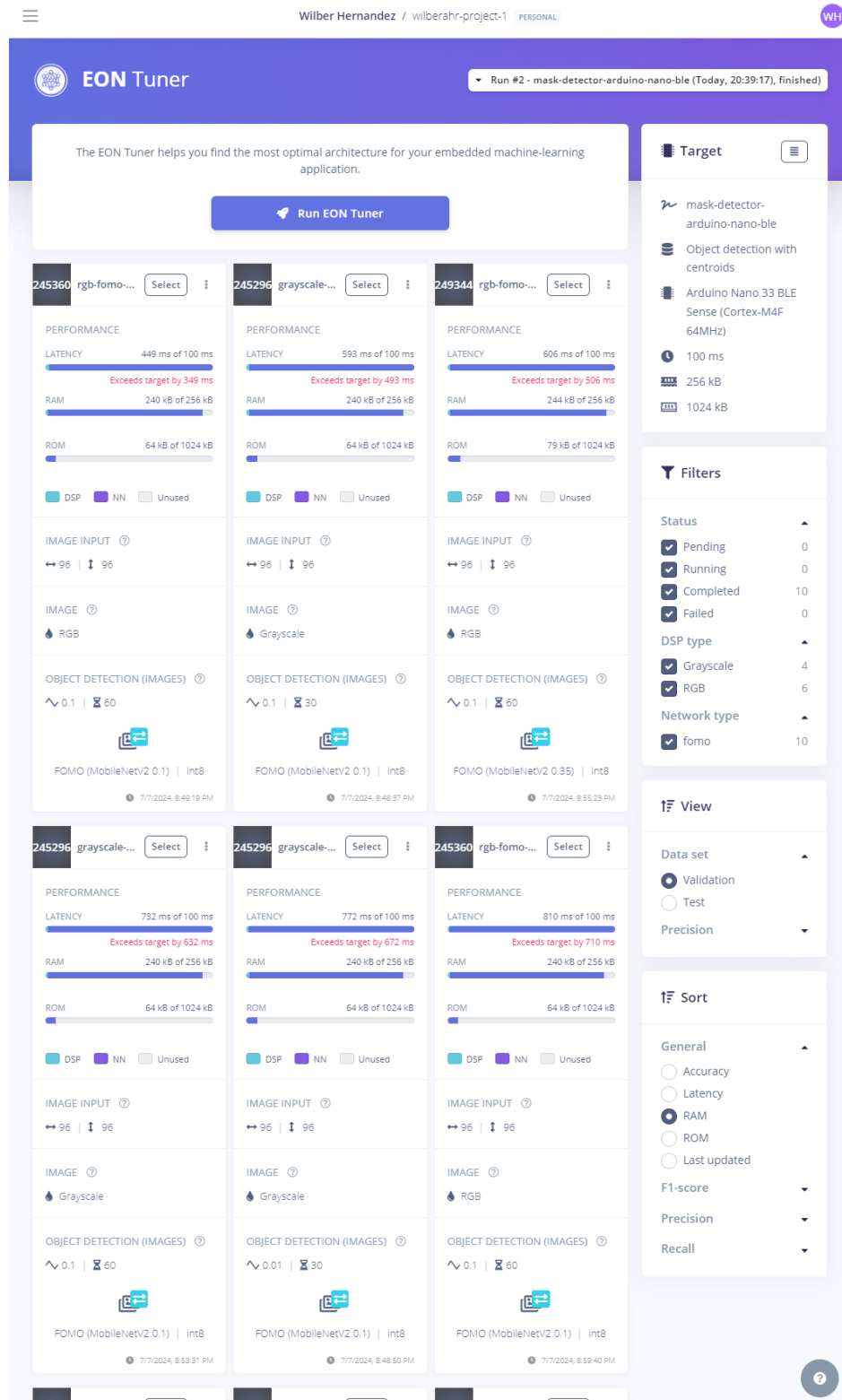


Figura 10: Búsqueda de la arquitectura óptima para la aplicación mediante Eonturner.

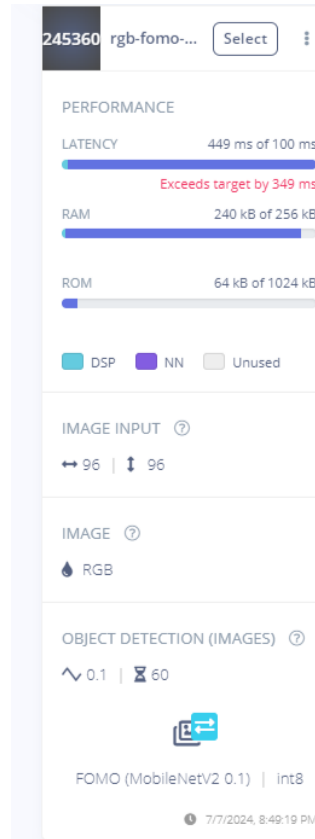


Figura 11: Selección de la arquitectura óptima para la aplicación.

Se realizó la prueba del modelo con el 20 % del conjunto de datos restante. Se obtuvo una exactitud del 62,50 % como se muestra en la figura 12.

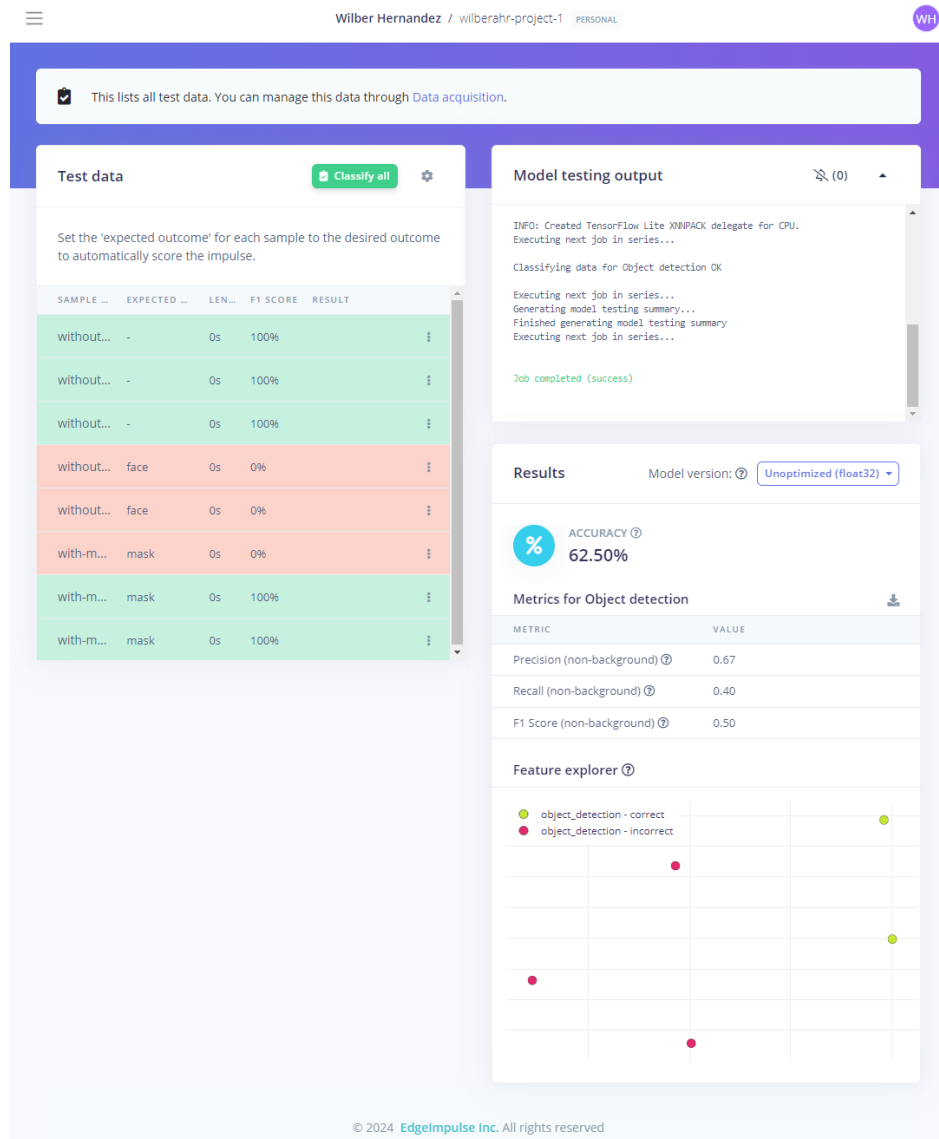



Figura 12: Generación de biblioteca de Arduino.

Finalmente, se generó la biblioteca de Arduino con el modelo a implementar en el Arduino Nano BLE 33. Se muestra la configuración de opciones en la figura 13.




SELECTED DEPLOYMENT

Arduino library

An Arduino library with examples that runs on most Arm-based Arduino development boards.

MODEL OPTIMIZATIONS

Model optimizations can increase on-device performance but may reduce accuracy.



EON™ Compiler

Same accuracy, 17% less RAM, 34% less ROM.

Quantized (int8)

Selected ✓

	IMAGE	OBJECT DETECTION	TOTAL
LATENCY	-	439 ms.	439 ms.
RAM	-	235.6K	235.6K
FLASH	-	64.5K	-
ACCURACY			-

Unoptimized (float32)

Select

	IMAGE	OBJECT DETECTION	TOTAL
LATENCY	-	7,087 ms.	7,087 ms.
RAM	-	887.2K	887.2K
FLASH	-	67.1K	-
ACCURACY			62.50%

To compare model accuracy, run model testing for all available optimizations.

Run model testing

Figura 13: Generación de biblioteca de Arduino.

A partir de la biblioteca se compiló uno de los ejemplos pero al cargarlo al arduino no detectaba la imagen correctamente.

5. Conclusión

Se pudo hacer el entrenamiento del modelo a partir del uso de la plataforma de Edge Impulse y el conjunto de datos obtenidos mediante la cámara del Kit de Machine Learning de Arduino. Para un mejor modelo es necesario agregar un mayor conjunto de datos y diversificarlo con más personas y máscaras. Sin embargo la implementación en el Arduino no fue satisfactoria con la prueba del ejemplo proporcionado por la plataforma dentro de la biblioteca compilada.

6. Repositorio

El repositorio de este laboratorio se encuentra en el siguiente enlace:
<https://github.com/wilberahr/arduino-tensorflow-tinymml>.