

Track 7

Inteligencia Artificial Aplicada



C. TinyML Kit: Image Classification Hands-On

Prof. Marcelo José Rovai
rovai@unifei.edu.br

UNIFEI - Universidade Federal de Itajubá, Brazil



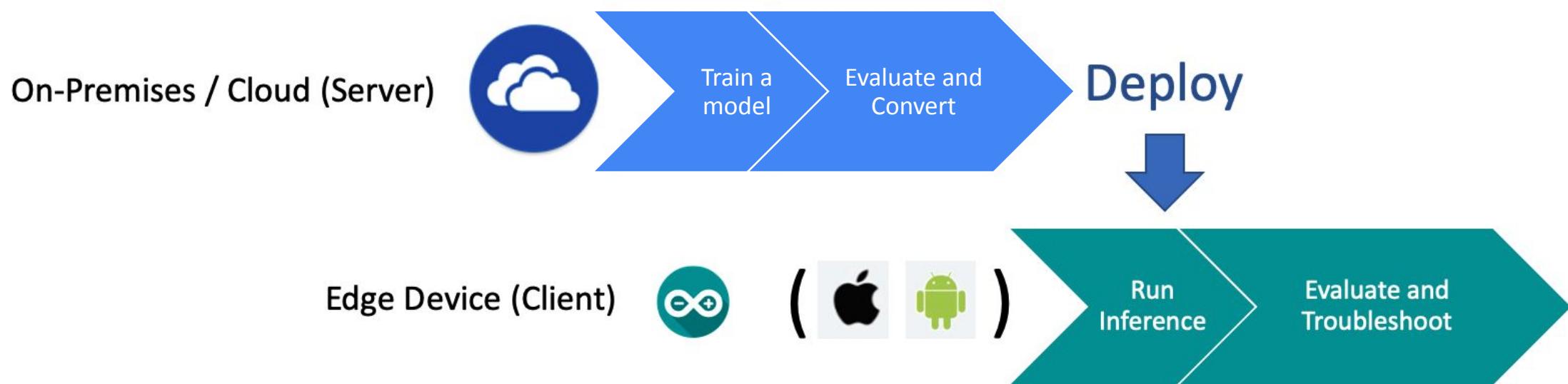
Image Classification Application: Design, Train, Test and Deploy

<https://studio.edgeimpulse.com/public/114253/latest>





* Feature Extraction





Select project - Edge Impulse

studio.edgeimpulse.com/studio/select-project

EDGE IMPULSE

Marcelo Rovai

Select project

Create a new project

Enter the name for your new project:

IESTI01 - Image Classification

Choose your project type:

Developer
20 min job limit, 4GB or 4 hours of data, limited collaboration.

Enterprise
No job or data size limits, higher performance, custom blocks. [Learn more](#)

Create new project

Marcelo Rovai / SciTinyML22-KWS

Marcelo Rovai / Cifar10_Image_Classification

Marcelo Rovai / IESTI01-Cifar10_Classification

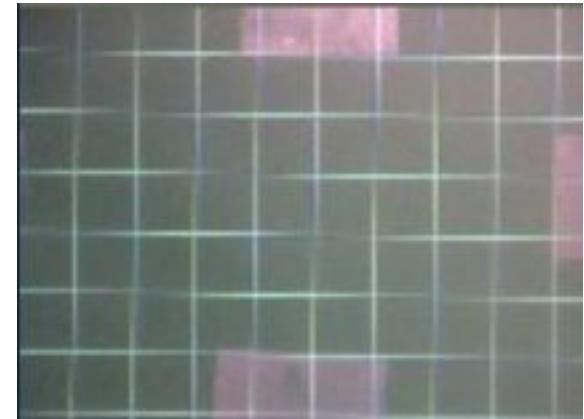
Marcelo Rovai / Bean Disease Classifier

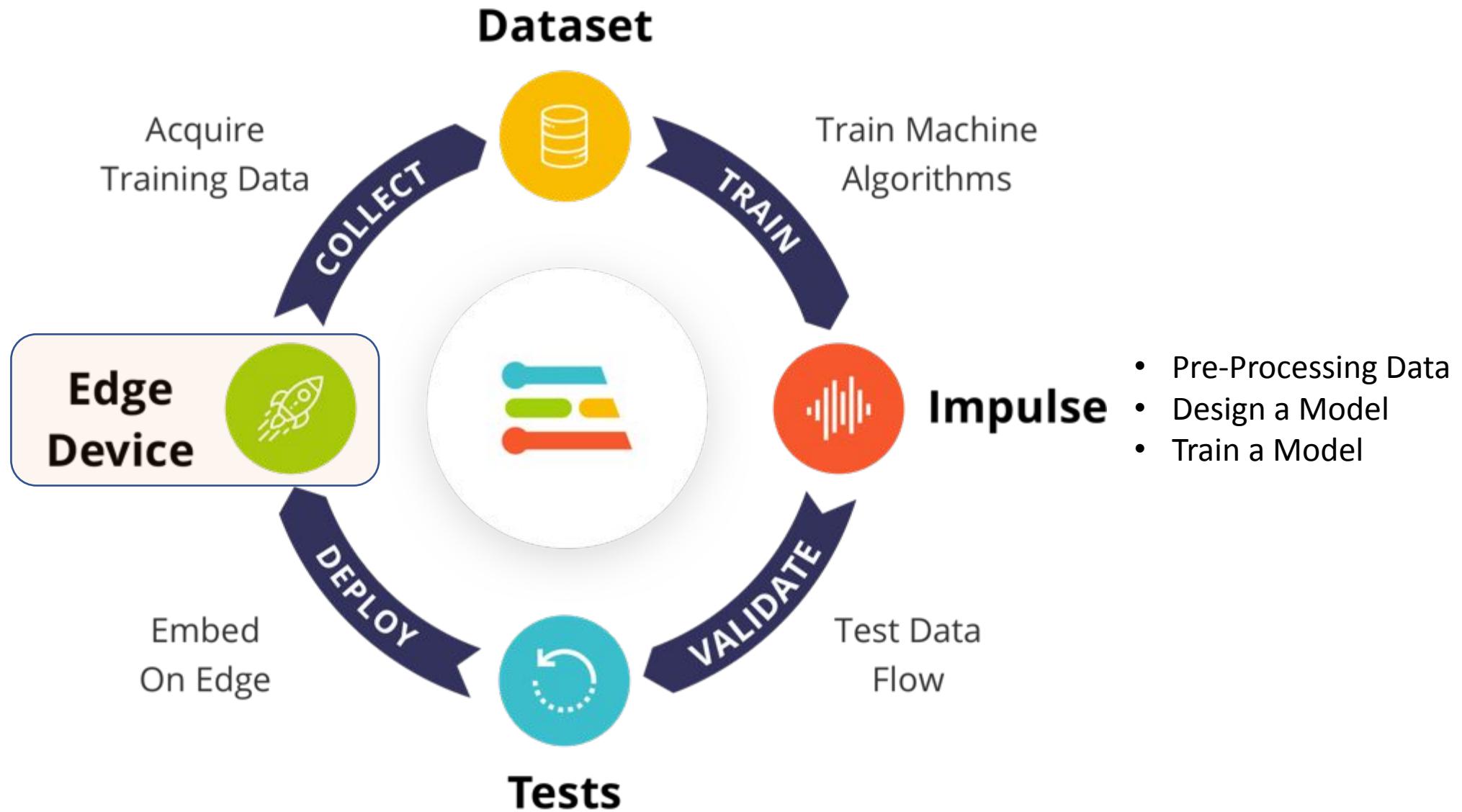
The screenshot shows the Edge Impulse Studio interface. A modal window titled "Create a new project" is open in the center. It prompts the user to "Enter the name for your new project:" with a text input field containing "IESTI01 - Image Classification". Below this, it asks "Choose your project type:" with two options: "Developer" (selected) and "Enterprise". The "Developer" option includes a note about a 20-minute job limit and 4GB/4-hour data limit. The "Enterprise" option includes a note about no job or data size limits and higher performance. At the bottom of the modal is a green "Create new project" button. In the background, there's a list of existing projects by Marcelo Rovai, each with a small profile picture next to the project name and a delete icon. The overall background has a grid pattern with orange and yellow squares.

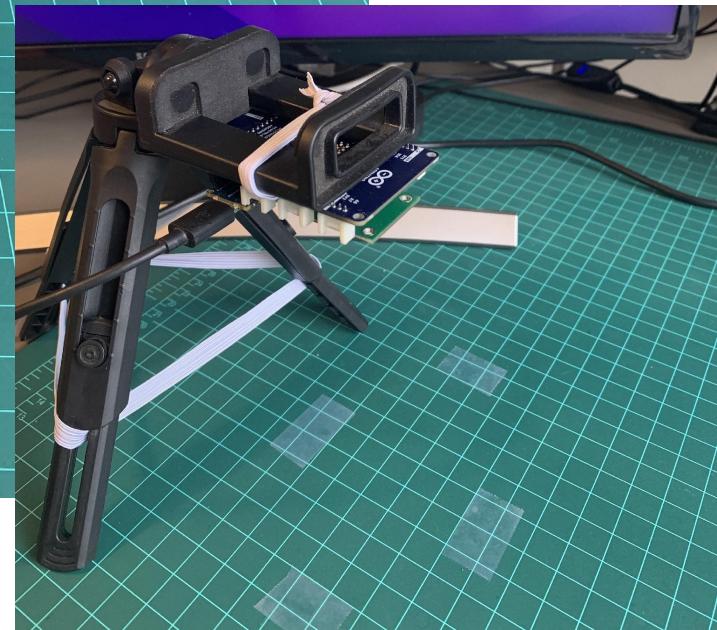
Image Classification Project

Decide a Goal

- Possible Images:
 - Medicine
 - background







EDGE IMPULSE Home API Reference Log In

Guides > Arduino Nano 33 BLE Sense Search

DOCUMENTATION

- Getting Started
- API and SDK references
- What is embedded ML, anyway?
- Frequently asked questions

DEVELOPMENT BOARDS

- Overview
- ST B-L475E-IOT01A
- Arduino Nano 33 BLE Sense**
- Eta Compute ECM3532 AI Sensor
- Eta Compute ECM3532 AI Vision
- OpenMV Cam H7 Plus
- Himax WE-I Plus
- Nordic Semi nRF52840 DK
- Nordic Semi nRF5340 DK
- SiLabs Thunderboard Sense 2
- Sony's Spresense
- Arduino Portenta H7 + Vision shield (preview)
- Raspberry Pi 4
- NVIDIA Jetson Nano
- Mobile phone
- Porting guide

COMMUNITY BOARDS

- Seeed Wio Terminal
- Agora Product Development Kit

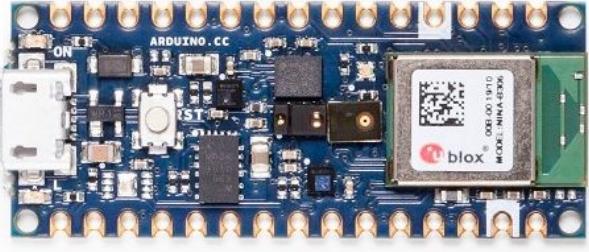
EDGE IMPULSE FOR LINUX

- Edge Impulse for Linux
- Linux Node.js SDK
- Linux Go SDK
- Linux C++ SDK
- Linux Python SDK

Arduino Nano 33 BLE Sense

The Arduino Nano 33 BLE Sense is a tiny development board with a Cortex-M4 microcontroller, motion sensors, a microphone and BLE - and it's fully supported by Edge Impulse. You'll be able to sample raw data, build models, and deploy trained machine learning models directly from the studio. It's available for around 30 USD from [Arduino](#) and a wide range of distributors.

The Edge Impulse firmware for this development board is open source and hosted on GitHub: [edgeimpulse/firmware-arduino-nano-33-ble-sense](#).



Arduino Nano 33 BLE Sense

Installing dependencies

To set this device up in Edge Impulse, you will need to install the following software:

1. [Edge Impulse CLI](#).
2. [Arduino CLI](#).
 - Here's an [instruction video for Windows](#).
 - The [Arduino website](#) has instructions for macOS and Linux.
3. On Linux:
 - GNU Screen: install for example via `sudo apt install screen`.

Note that the 1. Edge Impulse CLI is not necessary for Arduino Nano-33 if you use WebUSB.

DOCUMENTATION

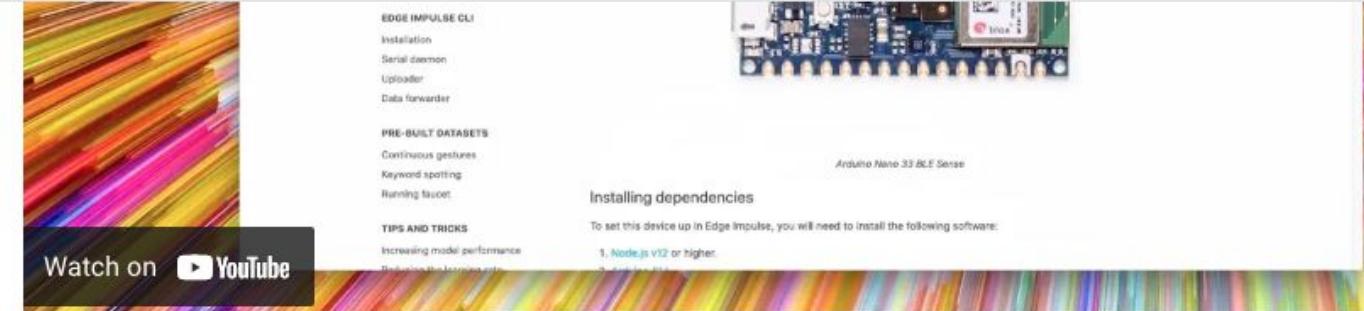
- Getting Started
- API and SDK references
- What is embedded ML, anyway?
- Frequently asked questions

DEVELOPMENT BOARDS

- Overview
- ST B-L475E-IOT01A
- Arduino Nano 33 BLE Sense**
- Eta Compute ECM3532 AI Sensor
- Eta Compute ECM3532 AI Vision
- OpenMV Cam H7 Plus
- Himax WE-I Plus
- Nordic Semi nRF52840 DK
- Nordic Semi nRF5340 DK
- SiLabs Thunderboard Sense 2
- Sony's Spresense
- Arduino Portenta H7 + Vision shield (preview)
- Raspberry Pi 4
- NVIDIA Jetson Nano
- Mobile phone
- Porting guide

COMMUNITY BOARDS

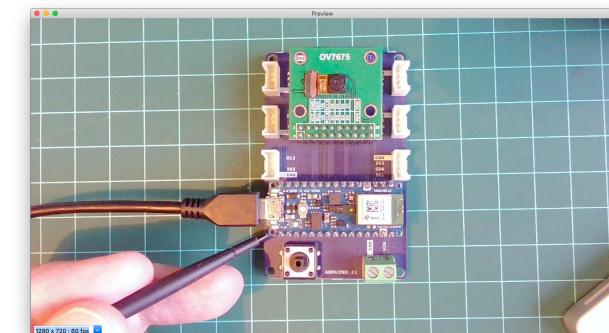
- Seeed Wio Terminal
- Agora Product Development Kit



The screenshot shows the Edge Impulse website with the Arduino Nano 33 BLE Sense development board highlighted. The page includes sections for the Edge Impulse CLI, Pre-built Databases, and Tips & Tricks. A 'Watch on YouTube' button is visible.

1. Connect the development board to your computer

Use a micro-USB cable to connect the development board to your computer. Then press RESET twice to launch into the bootloader. The on-board LED should start pulsating to indicate this.



Press RESET twice quickly to launch the bootloader on the Arduino Nano 33 BLE Sense.

2. Update the firmware

The development board does not come with the right firmware yet. To update the firmware:

1. [Download the latest Edge Impulse firmware](#), and unzip the file.
2. Open the flash script for your operating system (`flash_windows.bat`, `flash_mac.command` or `flash_linux.sh`) to flash the firmware.
3. Wait until flashing is complete, and press the RESET button once to launch the new firmware.

3. Setting keys

DOCUMENTATION

- Getting Started
- API and SDK references
- What is embedded ML, anyway?
- Frequently asked questions

DEVELOPMENT BOARDS

- Overview

ST B-L475E-IOT01A

Arduino Nano 33 BLE Sense

Eta Compute ECM3532 AI Sensor

Eta Compute ECM3532 AI Vision

OpenMV Cam H7 Plus

Himax WE-I Plus

Nordic Semi nRF52840 DK

Nordic Semi nRF5340 DK

SiLabs Thunderboard Sense 2

Sony's SpreSense

Arduino Portenta H7 + Vision shield
(preview)

Raspberry Pi 4

NVIDIA Jetson Nano

Mobile phone

Porting guide

COMMUNITY BOARDS

Seeed Wio Terminal

Agora Product Development Kit

**1. Connect the development board to your computer**

Use a micro-USB cable to connect the development board to your computer. Then press RESET twice to launch into the bootloader. The on-board LED should start pulsating to indicate this.

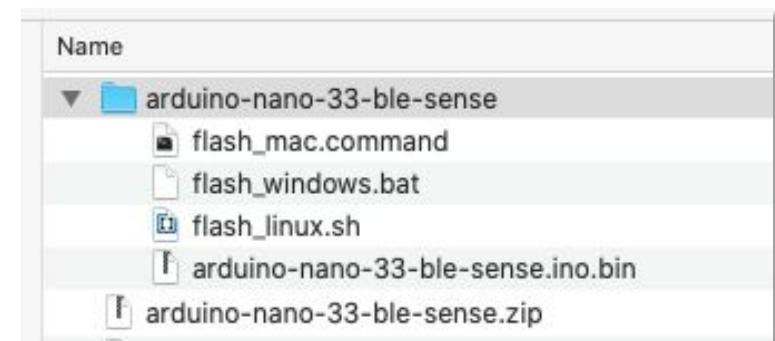
**1**

Press RESET twice quickly to launch the bootloader on the Arduino Nano 33 BLE Sense.

2. Update the firmware

The development board does not come with the right firmware yet. To update the firmware:

1. [Download the latest Edge Impulse firmware](#), and unzip the file.
2. Open the flash script for your operating system (`flash_windows.bat` , `flash_mac.command` or `flash_linux.sh`) to flash the firmware.
3. Wait until flashing is complete, and press the RESET button once to launch the new firmware.

3. Setting keys

MacOS

```
mjrovai — flash_mac.command — 124x43
Last login: Mon Jun 28 08:58:22 on ttys002
You have new mail.
/Users/mjrovai/Downloads/arduino-nano-33-ble-sense/flash_mac.command ; exit;

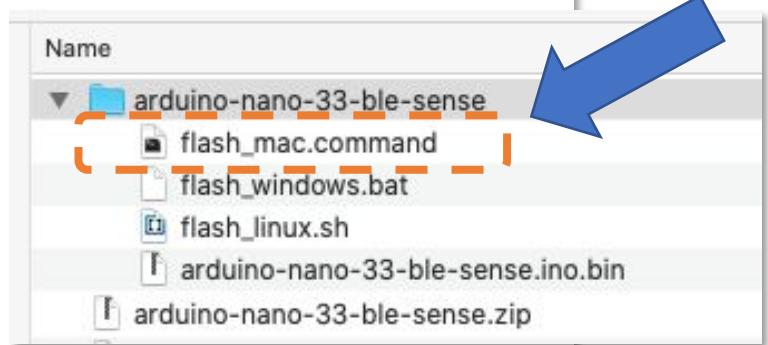
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) MacBook-Pro-de-Marcelo:~ mjrovai$ /Users/mjrovai/Downloads/arduino-nano-33-ble-sense/flash_mac.command ; exit;
Finding Arduino Mbed core...
Finding Arduino Mbed OK
Finding Arduino Nano 33 BLE...
Finding Arduino Nano 33 BLE OK
Flashing board...
Device      : nRF52840-QIAA
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 525440 bytes to flash (129 pages)
[=====] 100% (129/129 pages)
Done in 20.533 seconds

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'
logout
Saving session...
...copying shared history...
...saving history...truncating history files...
...completed.

[Process completed]
```

1. Press Nano-33 Reset button Twice
2. With Nano-33 LED Flashing:



3.Nano-33 LED Stop Flashing

Windows 10

```
Prompt de Comando
Microsoft Windows [versão 10.0.19041.1052]
(c) Microsoft Corporation. Todos os direitos reservados.

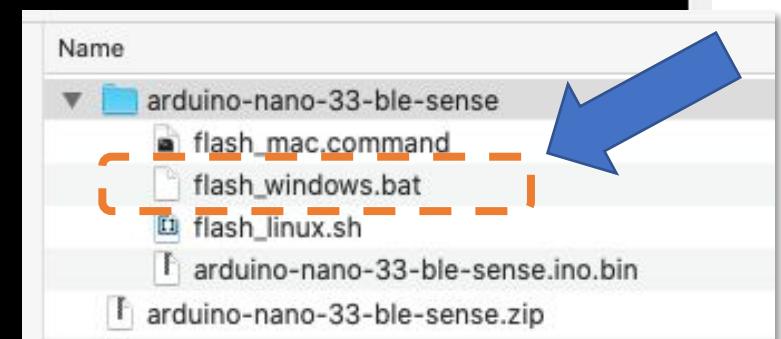
C:\Users\GUILH>arduino-cli
Arduino Command Line Interface (arduino-cli).

Usage:
  arduino-cli [command]

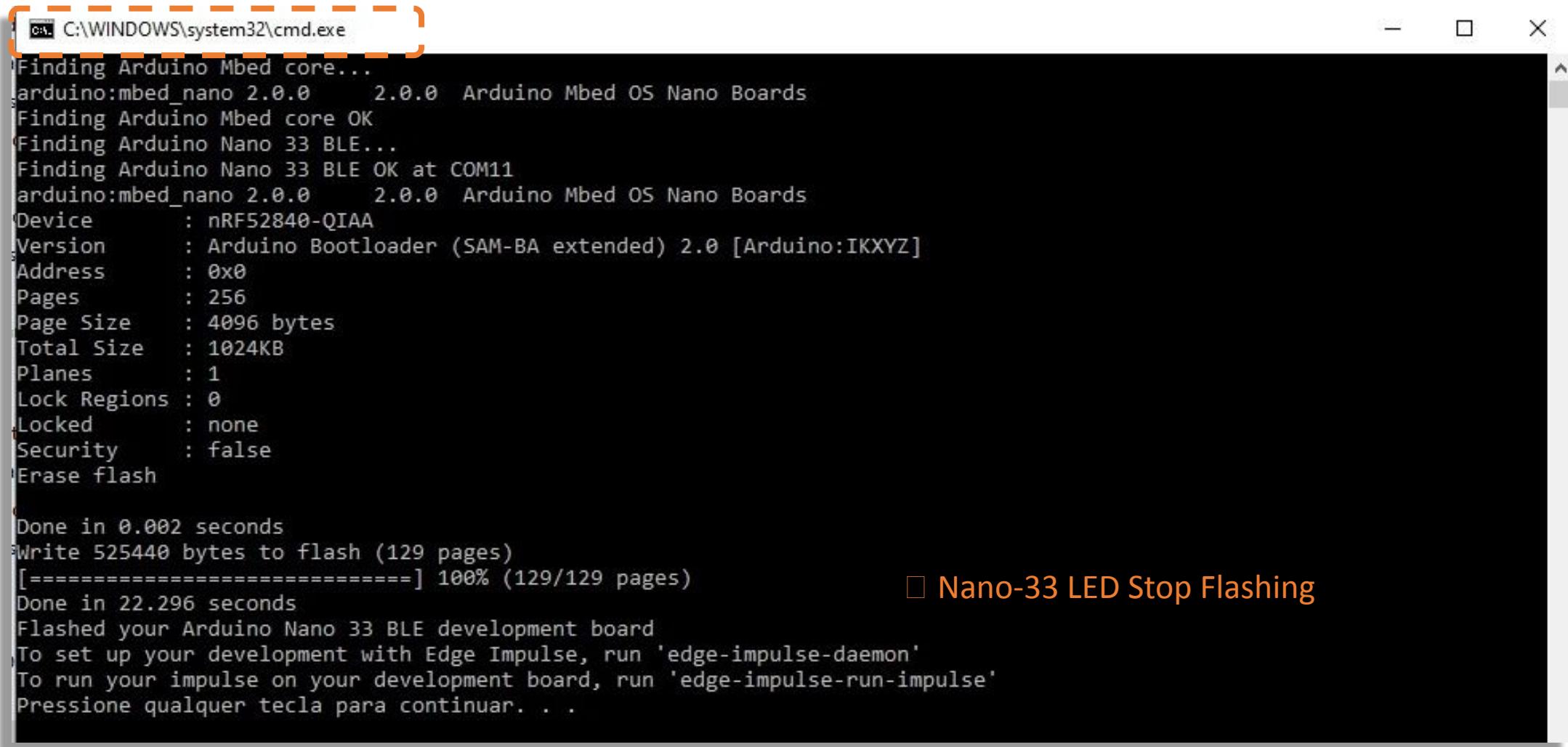
Examples:
  arduino-cli <command> [flags...]

Available Commands:
  board           Arduino board commands.
  burn-bootloader Upload the bootloader.
  cache           Arduino cache commands.
  compile         Compiles Arduino sketches.
  completion     Generates completion scripts
  config          Arduino configuration commands.
  core            Arduino core operations.
  daemon          Run as a daemon on port 50051
  debug           Debug Arduino sketches.
  help            Help about any command
  lib              Arduino commands about libraries.
  outdated        Lists cores and libraries that can be upgraded
  sketch          Arduino CLI sketch commands.
  update          Updates the index of cores and libraries
  upgrade         Upgrades installed cores and libraries.
  upload          Upload Arduino sketches.
  version         Shows version number of Arduino CLI.
```

1. Press Nano-33 Reset button Twice
2. With Nano-33 LED Flashing:



Windows 10



```
C:\WINDOWS\system32\cmd.exe
Finding Arduino Mbed core...
arduino:mbed_nano 2.0.0      2.0.0  Arduino Mbed OS Nano Boards
Finding Arduino Mbed core OK
Finding Arduino Nano 33 BLE...
Finding Arduino Nano 33 BLE OK at COM11
arduino:mbed_nano 2.0.0      2.0.0  Arduino Mbed OS Nano Boards
Device      : nRF52840-QIAA
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 0.002 seconds
Write 525440 bytes to flash (129 pages)
[=====] 100% (129/129 pages)
Done in 22.296 seconds
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'
Pressione qualquer tecla para continuar. . .
```

Nano-33 LED Stop Flashing

Follow this [video](#) to install the Arduino CLI on Windows 10 and add the 'arduino-cli' binary to your PATH. **This makes the CLI available from any location through the command prompt.**

The screenshot shows the Edge Impulse Studio interface. On the left, a sidebar lists project categories: EDG, Dashboard, Devices, Data acquisition, ImpulseNet, Create, Retrain, Live classifier, Model testing, Versioning, Deployment, Documentation, and Forums. A central modal dialog titled "studio.edgeimpulse.com wants to connect to a serial port" lists available ports: cu.Bluetooth-Incoming-Port, cu.MALS, cu.RovaisAirPods-Wireless, cu.SOC, and Nano 33 BLE (cu.usbmodem144301) - Paired. The "Nano 33 BLE" port is selected and highlighted in blue. At the bottom of the dialog are "Cancel" and "Connect" buttons, with the "Connect" button highlighted by a red dashed box and labeled "3". The main workspace shows a purple header bar with the project name "POTTING_PROJECT" and a user profile for "MJRoBot (Marcelo Rovai)". Below the header is a message: "Record data from any device or development board, or upload your existing datasets - Show options". The central area displays a chart with "LABELS 0" and a message "No data collected yet". A green button at the bottom right says "Let's collect some data". To the right, a section titled "Record new data" shows a message "2" and a "Connect using WebUSB" button, also highlighted by a red dashed box and labeled "2". Below this is a dark panel with the text "RAW DATA" and "Click on a sample to load...".

Do not forget to connect the Nano with your PC serial via Arduino-CLI, before starting collecting data.

1 → 2 → 3



Data acquisition - IESTI01

studio.edgeimpulse.com/studio/114253/acquisition/training?page=1

Record new data

Device ②
36:17:55:F9:70:F7

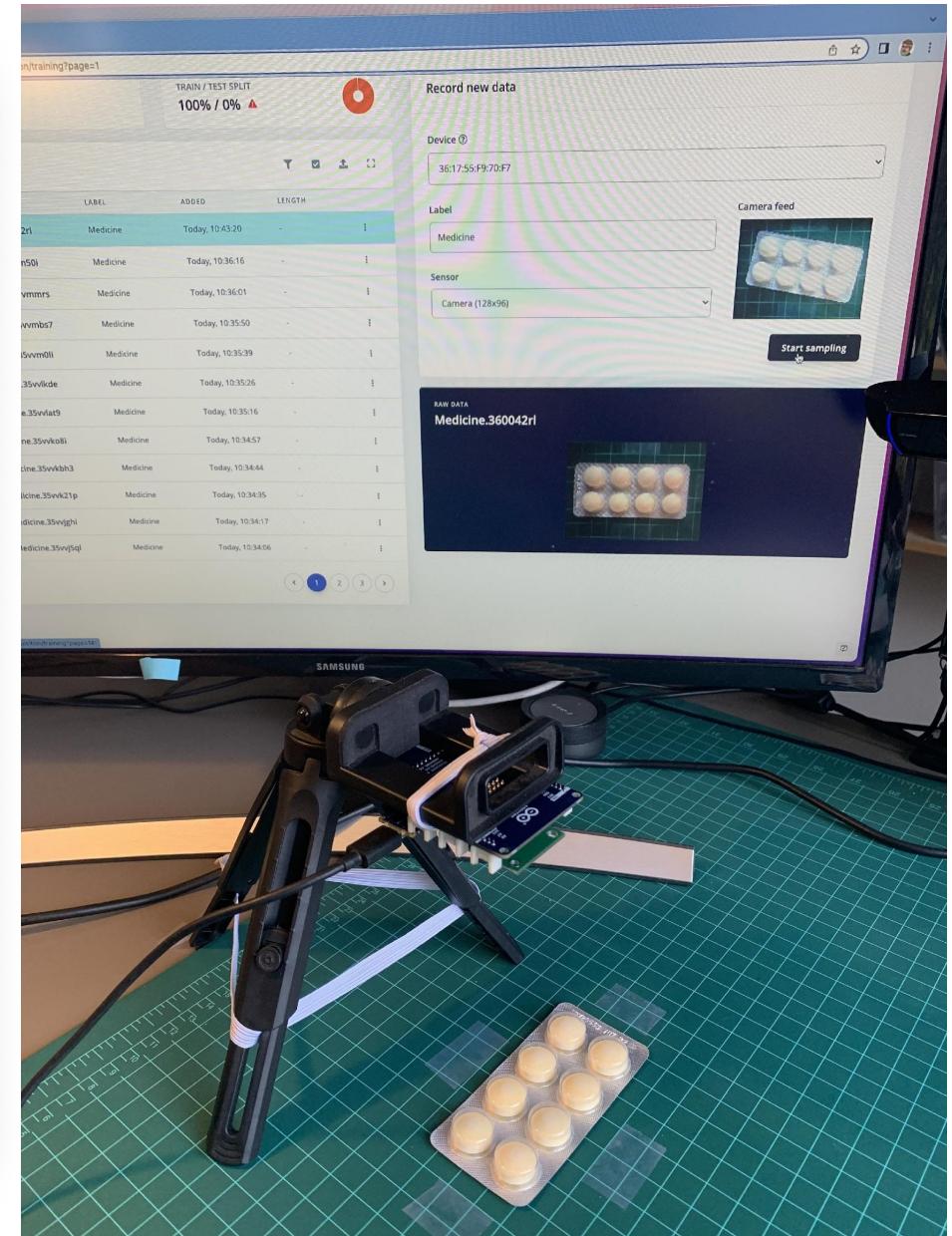
Label
medicine

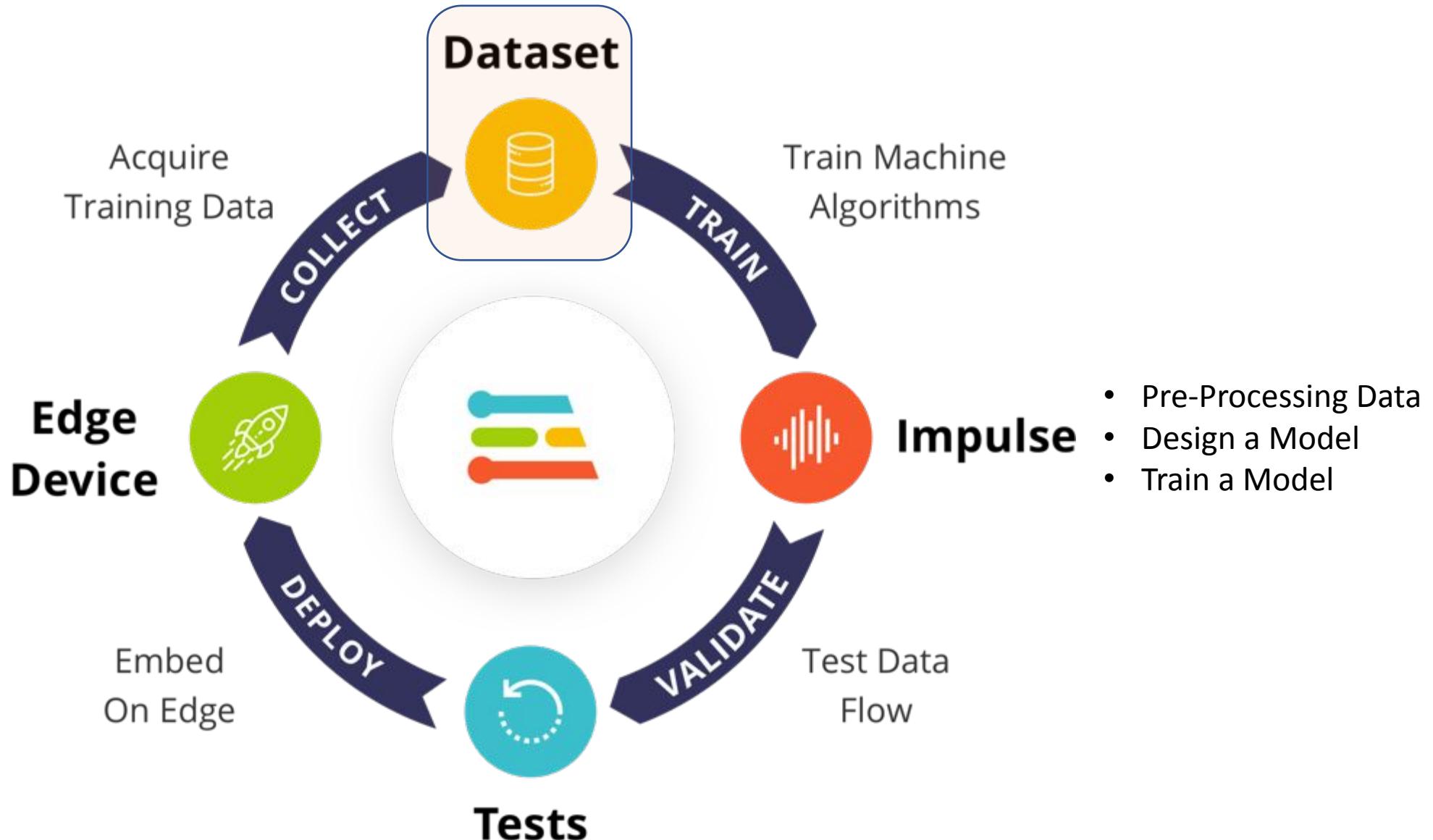
Sensor
Camera (128x96)

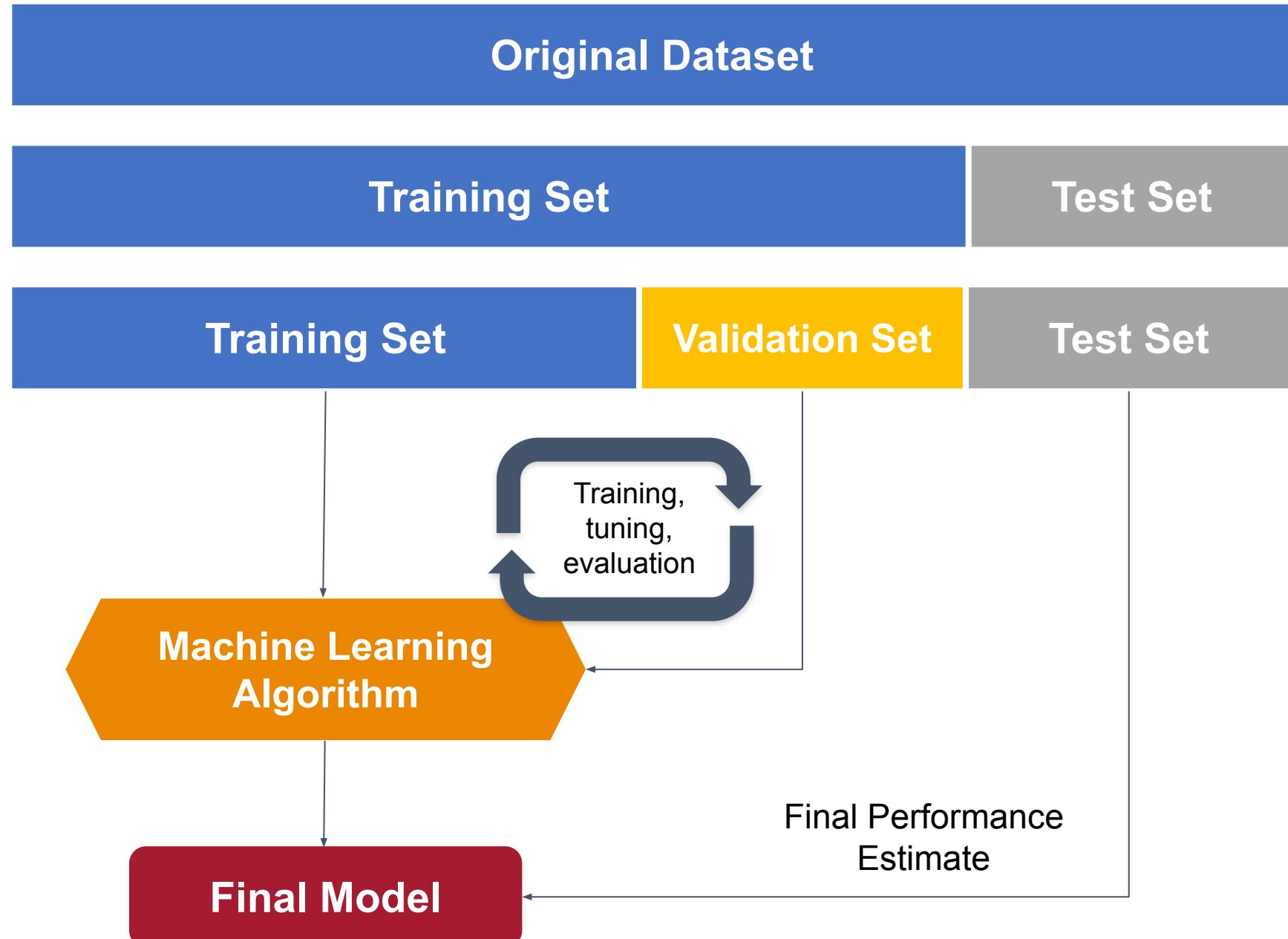
Camera feed

Start sampling

RAW DATA
Click on a sample to load...







Data acquisition - IESTI01 - Image classification

studio.edgeimpulse.com/studio/114253/acquisition/training?page=5

EDGE IMPULSE

DATA ACQUISITION (IESTI01 - IMAGE CLASSIFICATION)

Training data Test data | Data explorer | Upload data Export data

Did you know? You can capture data from any device or development board, or upload your existing datasets - Show options

DATA COLLECTED 68 items

TRAIN / TEST SPLIT 85% / 15%

Collected data

SAMPLE NAME	LABEL	ADDED	LENGTH
Medicine.35vvihal	medicine	Today, 10:33:45	-
Medicine.35vvi6da	medicine	Today, 10:33:34	-
Medicine.35vvho70	medicine	Today, 10:33:19	-
Medicine.35vvhep2	medicine	Today, 10:33:09	-
Medicine.35vh3qf	medicine	Today, 10:32:58	-
Medicine.35vvgtfs	medicine	Today, 10:32:52	-
Medicine.35vvgn6j	medicine	Today, 10:32:45	-
Medicine.35vgcbr	medicine	Today, 10:32:34	-
Medicine.35vvg2s0	medicine	Today, 10:32:24	-
Medicine.35vvfith	medicine	Today, 10:32:08	-
Medicine.35v vf6ij	medicine	Today, 10:31:55	-
Medicine.35vver8u	medicine	Today, 10:31:44	-

Record new data Connect using WebUSB

No devices connected to the remote management API.

RAW DATA Medicine.35vvi6da



19

DATA ACQUISITION (IESTI01 - IMAGE CLASSIFICATION)

Training data Test data | Data explorer | Upload data Export data

Did you know? You can capture data from any device or development board, or upload your existing datasets - Show options

DATA COLLECTED 68 items TRAIN / TEST SPLIT 85% / 15%

Collected data

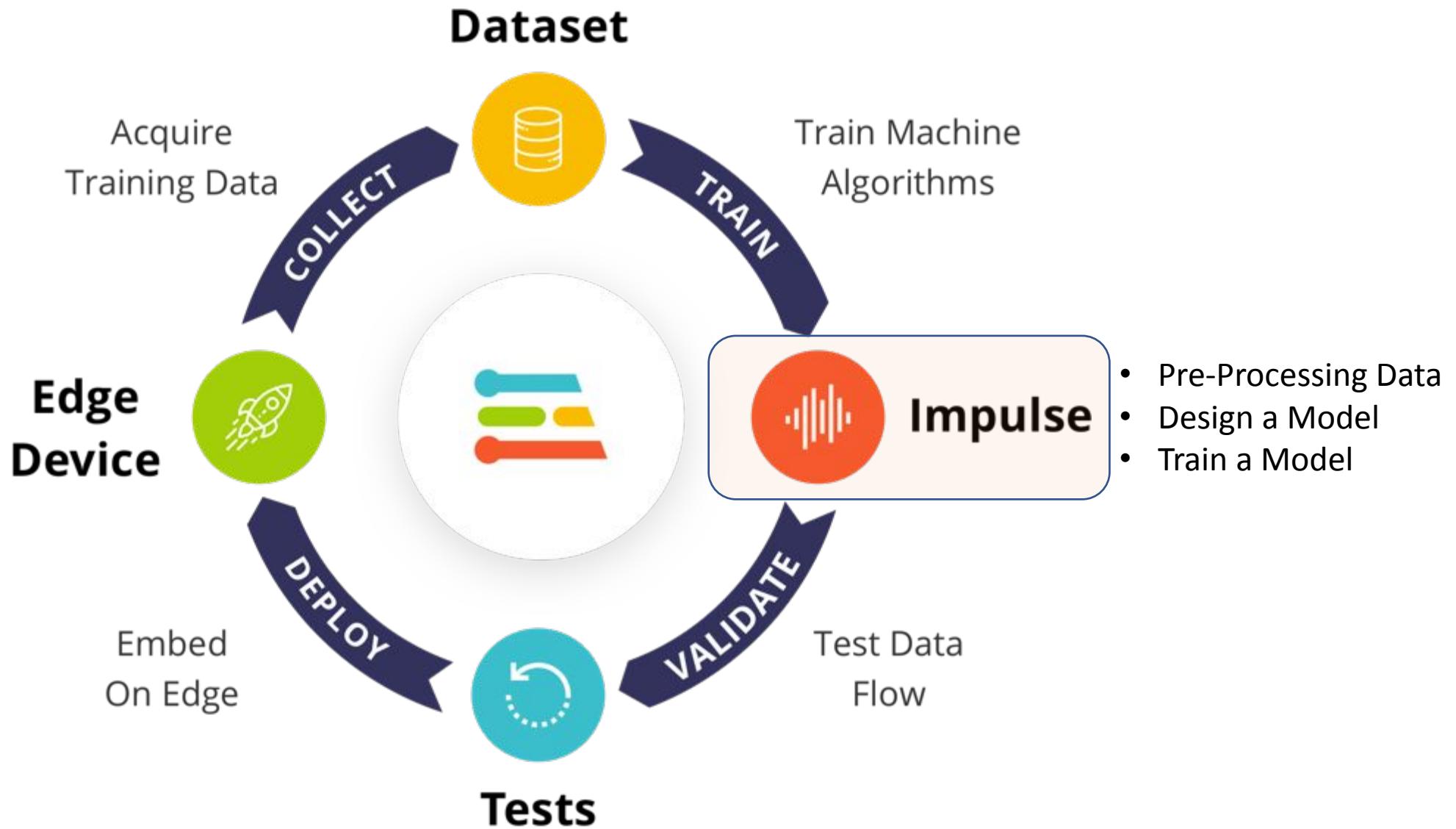
SAMPLE NAME	LABEL	ADDED	LENGTH
background.3600e7fj	background	Today, 10:48:52	-
background.3600e161	background	Today, 10:48:46	-
background.3600dqt1	background	Today, 10:48:39	-
background.3600dj23	background	Today, 10:48:31	-
background.3600dc09	background	Today, 10:48:25	-
background.3600d7o5	background	Today, 10:48:20	-
background.3600d38u	background	Today, 10:48:15	-
background.3600cjir	background	Today, 10:47:59	-
background.3600cdb7	background	Today, 10:47:53	-
background.3600c5e5	background	Today, 10:47:45	-
background.3600bv6a	background	Today, 10:47:38	-
background.3600bnb4	background	Today, 10:47:30	-

Record new data Connect using WebUSB

No devices connected to the remote management API.

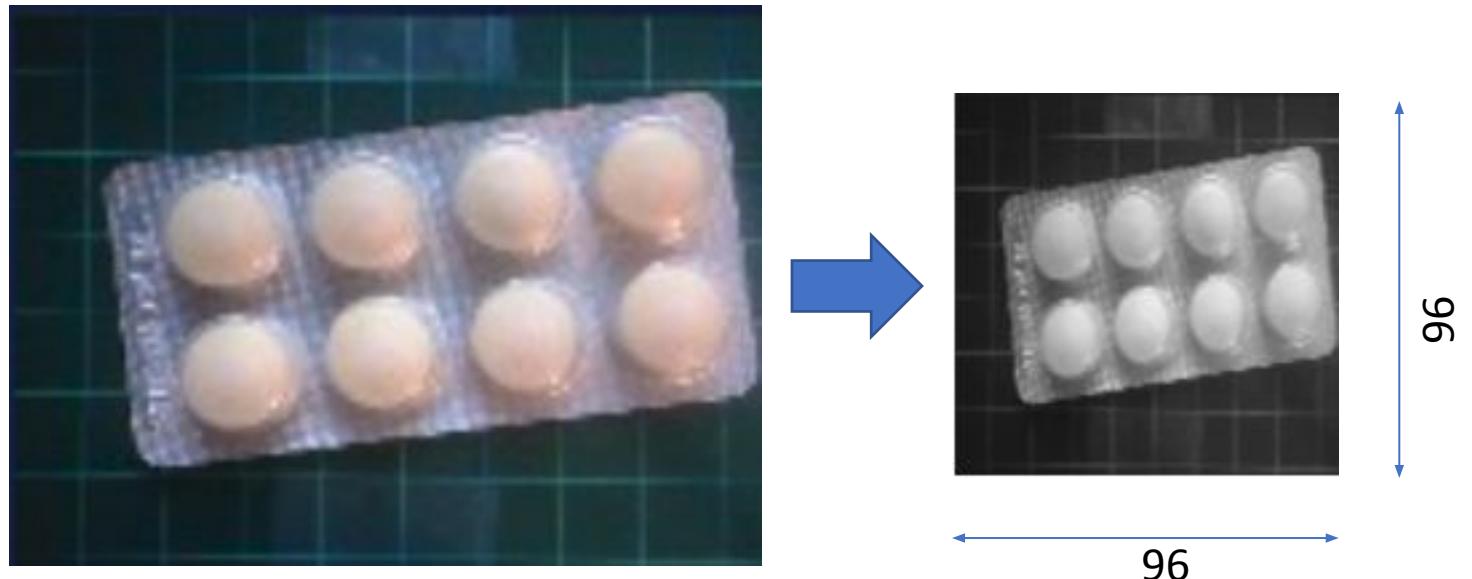
RAW DATA background.3600e161

1 2 3 4 5 6 >



Data Pre-processing

- Image Pre-Process *:
 - Convert to Grayscale
 - Re-scale 96 x 96



- * During Inference, the OV7675 captures the raw image as:
- QQVGA (160 x 120), crop it and resize it to 96x96
 - RGB565 and convert it first to RGB888 and after it to Grayscale
 - 1FPS

EDGE IMPULSE

CREATE IMPULSE (IESTI01 - IMAGE CLASSIFICATION)

An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.

Image data

Input axes
image

Image width 96 **Image height** 96

Resize mode
Squash

For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.

Image

Name Image

Input axes (1)
 image

Transfer Learning (Images)

Name Transfer learning

Input features
 Image

Output features
2 (background, medicine)

Output features

2 (background, medicine)

Save Impulse

Add a processing block

Add a learning block

Dashboard

Devices

Data sources

Data acquisition

Impulse design

Create impulse

Image

Transfer learning

EON Tuner

Retrain model

Live classification

Model testing

Versioning

Deployment

GETTING STARTED

Documentation

Forums

© 2022 EdgeImpulse Inc. All rights reserved

Marcelo Rovai

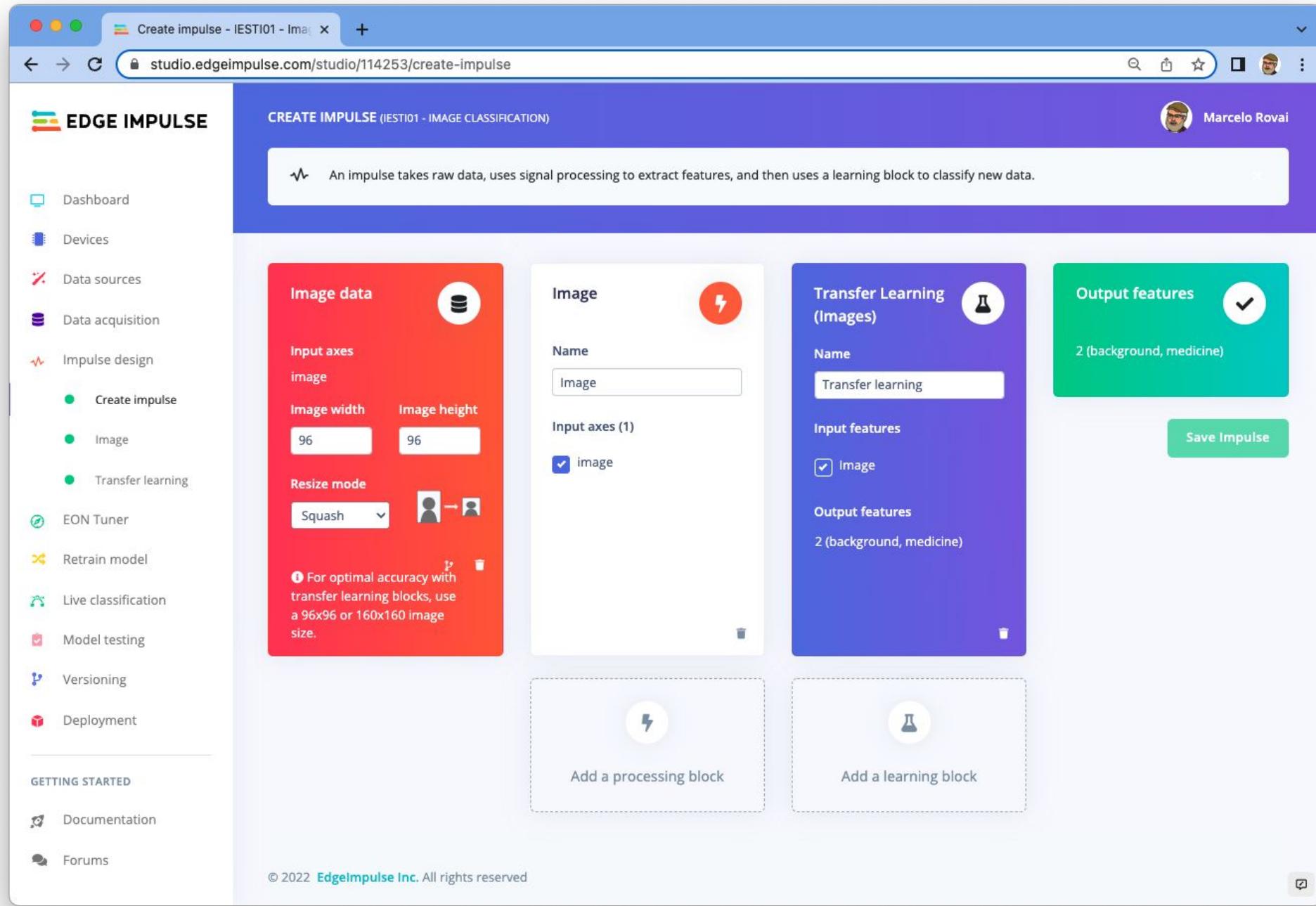


Image - IESTI01 - Image Classifi... +

studio.edgeimpulse.com/studio/114253/dsp/image/3

EDGE IMPULSE

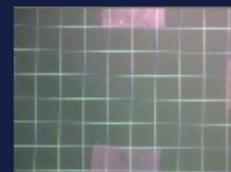
IMAGE (IESTI01 - IMAGE CLASSIFICATION)

#1 ▾ Click to set a description for this version

Parameters Generate features

Raw data

background.3600e7fj (background) ▾



Raw features ⋮

0x7b8376, 0x798174, 0x798174, 0x7a8072, 0x7b8072, 0x7e8274, 0x7d8173...

Parameters

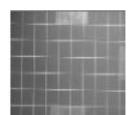
Image

Color depth Grayscale

Save parameters

DSP result

Image



Processed features ⋮

0.4985, 0.4907, 0.4907, 0.4887, 0.4898, 0.4989, 0.4949, 0.4942, 0.50...

On-device performance ?

PROCESSING TIME 11 ms.

PEAK RAM USAGE 4 KB

Image - IESTI01 - Image Classifi... +

studio.edgeimpulse.com/studio/114253/dsp/image/3/generate-features

EDGE IMPULSE

IMAGE (IESTI01 - IMAGE CLASSIFICATION)
#1 ▾ Click to set a description for this version

Parameters **Generate features**

Dashboard Devices Data sources Data acquisition Impulse design Create impulse Image Transfer learning EON Tuner Retrain model Live classification Model testing Versioning Deployment

GETTING STARTED Documentation Forums

Training set

Data in training set 68 items
Classes 2 (background, medicine)

Generate features

Feature explorer

background medicine

On-device performance ⓘ

PROCESSING TIME 11 ms. PEAK RAM USAGE 4 KB

© 2022 EdgeImpulse Inc. All rights reserved

Marcelo Rovai

Model Design

MobileNetV1 96x96 0.25

A pre-trained multi-layer convolutional network designed to efficiently classify images. Uses around 105.9K RAM and 301.6K ROM with default settings and optimizations. Works best with 96x96 input size. Supports both RGB and grayscale.

Model	Image Size
-------	------------

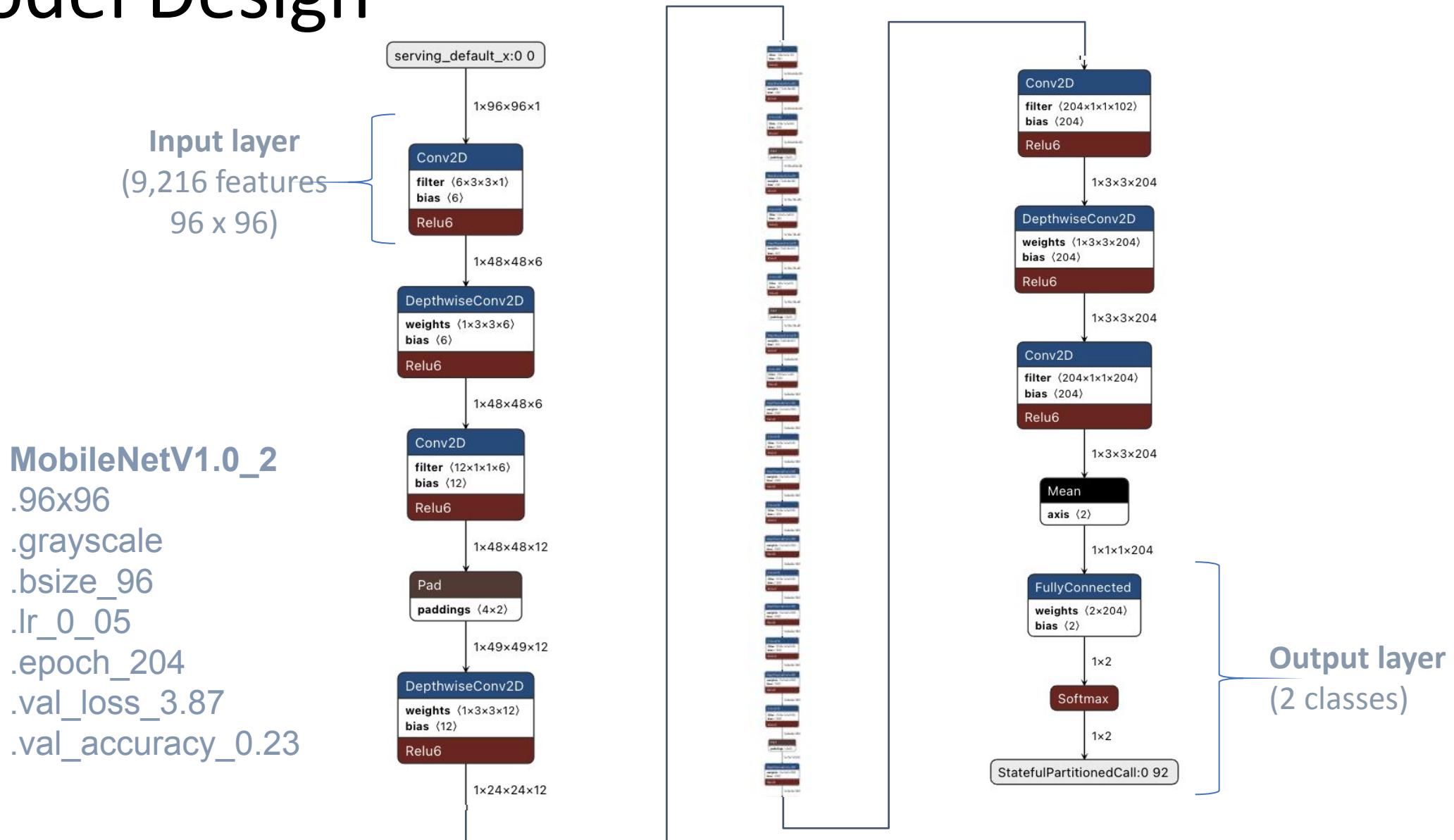
MobileNetV1	96x96 0.2	Alpha
-------------	-----------	-------

Uses around 83.1K RAM and 218.3K ROM with default settings and optimizations. Works best with 96x96 input size. Supports both RGB and grayscale.

MobileNetV1 96x96 0.1

Uses around 53.2K RAM and 101K ROM with default settings and optimizations. Works best with 96x96 input size. Supports both RGB and grayscale.

Model Design



Train

Transfer learning - IESTI01 - In progress

studio.edgeimpulse.com/studio/114253/learning/keras-transfer-image/5

EDGE IMPULSE

TRANSFER LEARNING (IESTI01 - IMAGE CLASSIFICATION)
#1 Click to set a description for this version

Dashboard

Devices

Data sources

Data acquisition

Impulse design

- Create impulse
- Image
- Transfer learning

EON Tuner

Retrain model

Live classification

Model testing

Versioning

Deployment

GETTING STARTED

Documentation

Forums

Neural Network settings

Training settings

- Number of training cycles: 50
- Learning rate: 0.0005
- Validation set size: 20 %
- Auto-balance dataset:
- Data augmentation:

Neural network architecture

- Input layer (9,216 features)
- MobileNetV1 96x96 0.2 (no final dense layer, 0.1 dropout)
- Choose a different model
- Output layer (2 classes)

Start training

High Latency

Training output

Very Good Accuracy

Model
Model version: Quantized (int8)

Last training performance (validation set)

ACCURACY 100.0%	LOSS 0.28
-----------------	-----------

Confusion matrix (validation set)

	BACKGROUND	MEDICINE
BACKGROUND	100%	0%
MEDICINE	0%	100%
F1 SCORE	1.00	1.00

Data explorer (full training set)

background - correct

medicine - correct

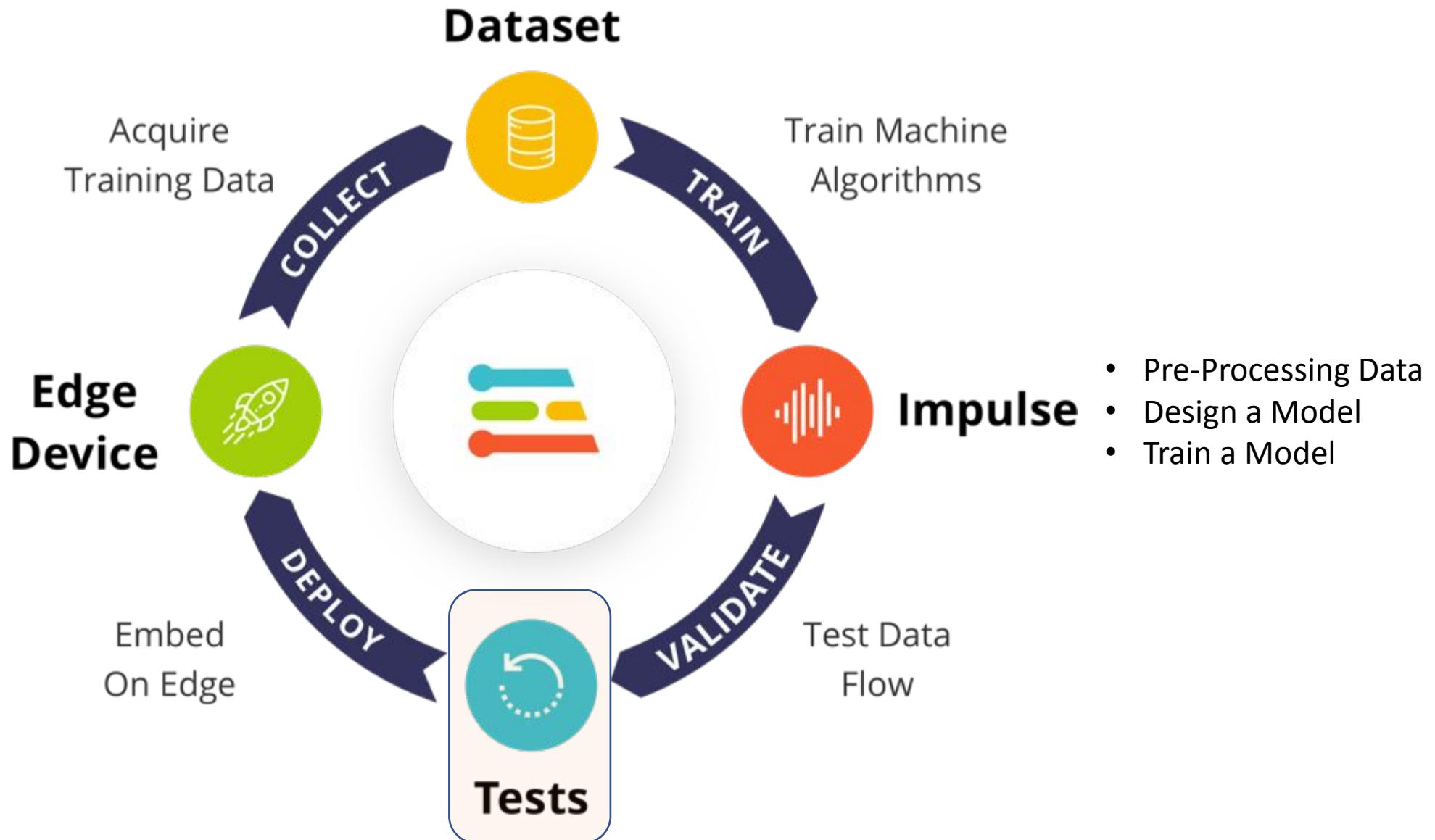
medicine - incorrect

On-device performance

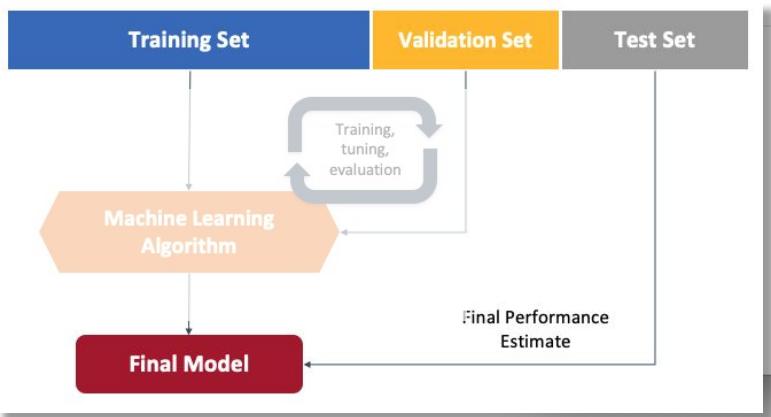
High Latency

Memory OK

INFERENCING TIME: 3,592 ms.	PEAK RAM USED: 106.2K	FLASH USAGE: 225.3K
-----------------------------	-----------------------	---------------------



Test



The diagram illustrates the machine learning workflow:

- Training Set**, **Validation Set**, and **Test Set** are input into the **Machine Learning Algorithm**.
- The **Machine Learning Algorithm** processes the data and outputs the **Final Model**.
- A feedback loop labeled "Training, tuning, evaluation" connects the **Validation Set** back to the **Machine Learning Algorithm**.
- The **Final Model** is used to estimate final performance.

EDGE IMPULSE

Test data

Set the 'expected outcome' for each sample to the desired outcome to automatically score the impulse.

SAMPLE ...	EXPECTED O...	LEN...	ACCURA...	RESULT	⋮
Medici...	background	-	100%	1 background	⋮
Medici...	background	-	100%	1 background	⋮
Medici...	medicine	-	100%	1 medicine	⋮
Medici...	medicine	-	100%	1 medicine	⋮
Medici...	medicine	-	100%	1 medicine	⋮
Medici...	medicine	-	100%	1 medicine	⋮
Medici...	medicine	-	100%	1 medicine	⋮
Medici...	medicine	-	100%	1 medicine	⋮
Medici...	medicine	-	100%	1 medicine	⋮
backgr...	background	-	100%	1 background	⋮
backgr...	background	-	100%	1 background	⋮
backgr...	background	-	100%	1 background	⋮

Model testing output

```
completed 200 / 500 epochs
completed 250 / 500 epochs
completed 300 / 500 epochs
completed 350 / 500 epochs
completed 400 / 500 epochs
completed 450 / 500 epochs
Tue Jun 21 15:13:32 2022 Finished embedding
Reducing dimensions for visualizations OK
Classifying data for Transfer learning...
Classifying data for float32 model...
Scheduling job in cluster...
Job started
Job completed
```

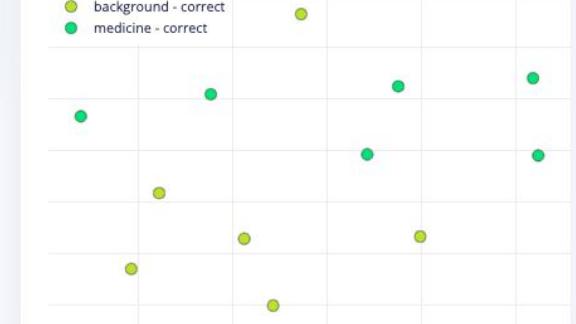
Model testing results

ACCURACY
100.00%

	BACKGROUND	MEDICINE	UNCERTAIN
BACKGROUND	100%	0%	0%
MEDICINE	0%	100%	0%
F1 SCORE	1.00	1.00	

Feature explorer

Legend: background - correct (yellow), medicine - correct (green)



The Feature explorer displays a scatter plot with points representing samples. The legend indicates that yellow points represent "background - correct" and green points represent "medicine - correct". The plot shows a clear separation between the two classes, with most points falling into their respective categories.

Classification result

Summary

Name	testing.3601u5mo
Expected outcome	testing
CATEGORY	COUNT
background	1
medicine	0
uncertain	0

Detailed result

Show only unknowns

BACKGROUND	MEDICINE
0.91	0.09

RAW DATA
testing.3601u5mo



Raw features 

0x92917f, 0x92917f, 0x93907f, 0x969080, 0x969080, 0x979080, 0x958e7e, 0x948e80, 0x948f83, 0x949085, 0x939085, 0x8...

Classification result

Summary

Name	testing.36020g3v
Expected outcome	testing
CATEGORY	COUNT
background	0
medicine	1
uncertain	0

Detailed result

Show only unknowns

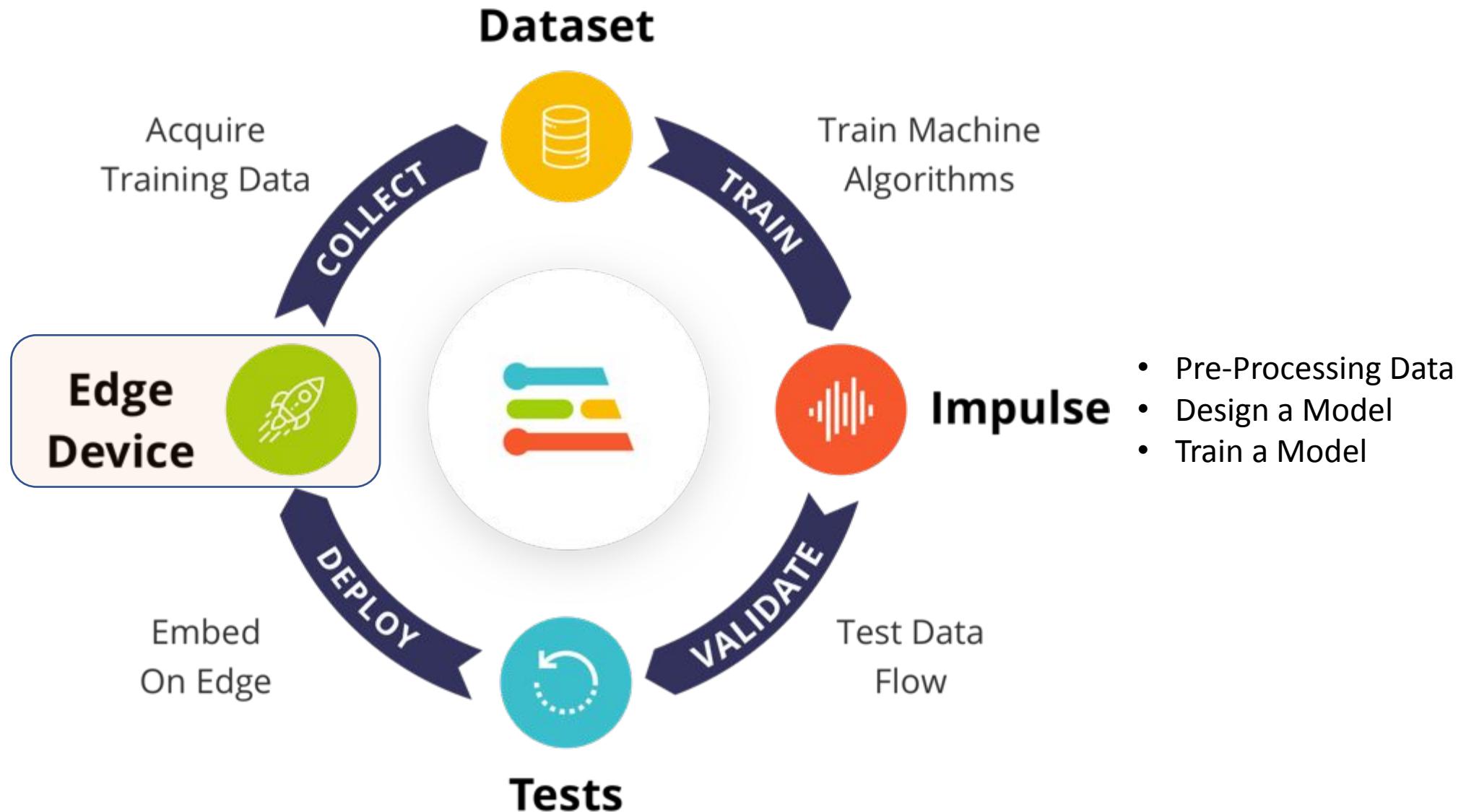
BACKGROUND	MEDICINE
0.03	0.97

RAW DATA
testing.36020g3v

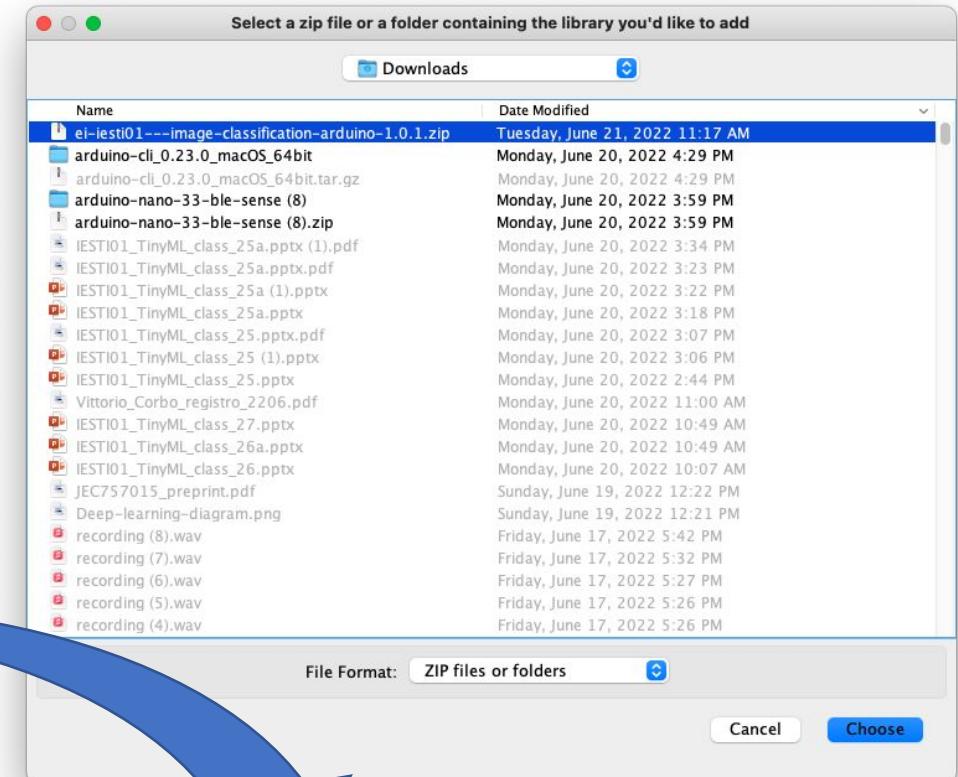


Raw features 

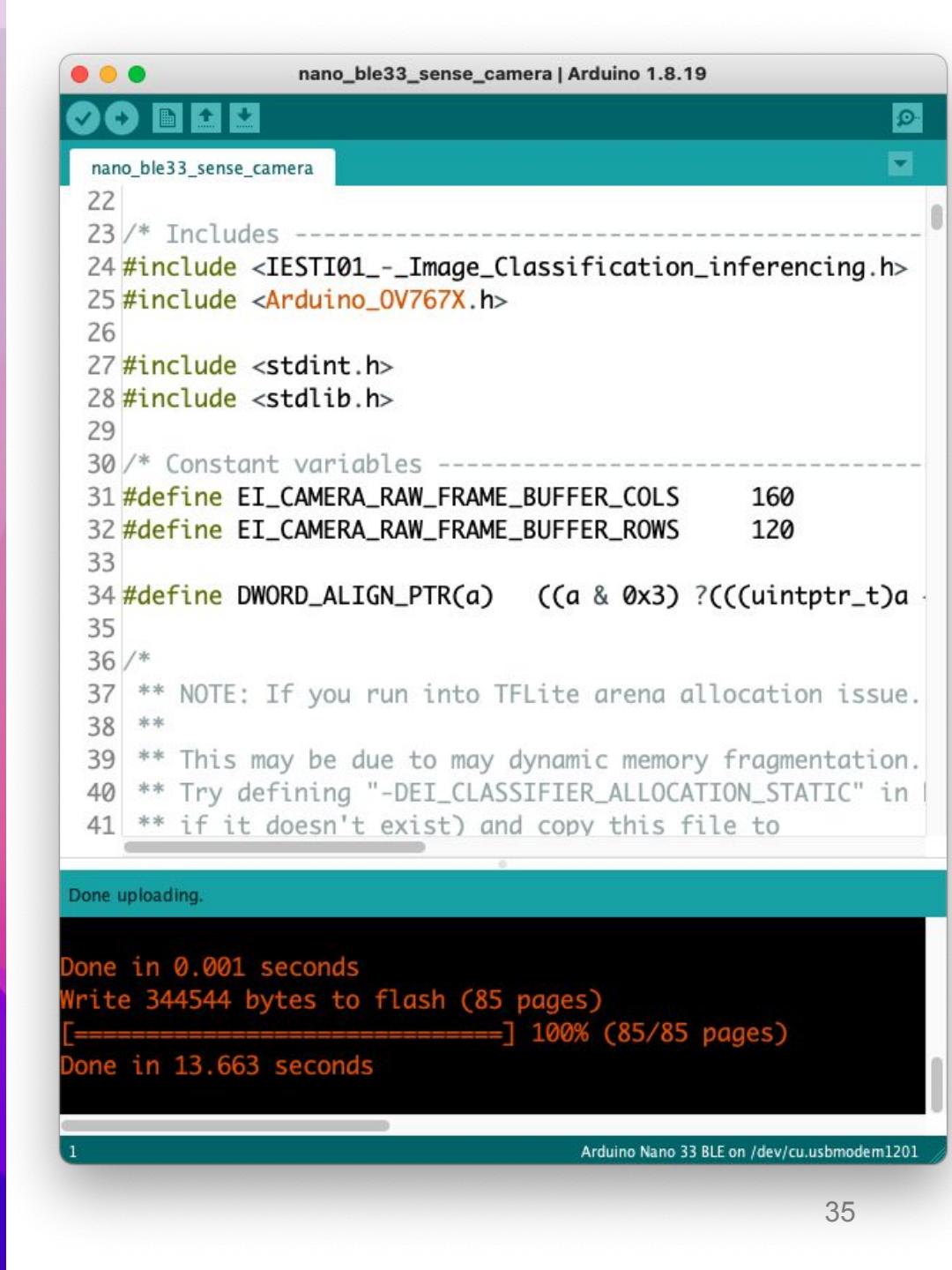
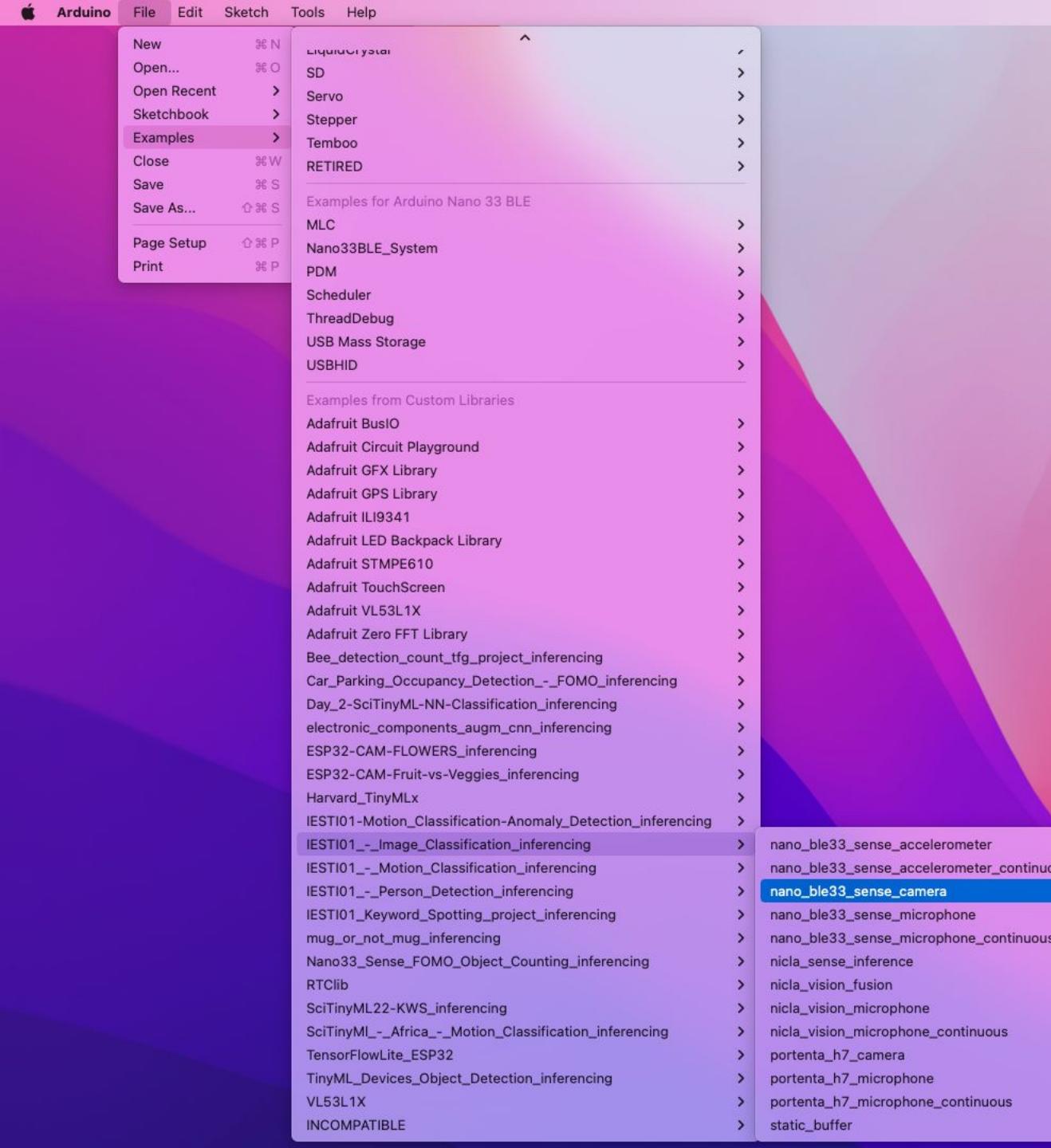
0x1d2a23, 0x1d2a23, 0x1c2922, 0x1d2824, 0x1e2825, 0x1e2925, 0x1f2a26, 0x1d2926, 0x1e2a28, 0x1c2b28, 0x1b2a28, 0x1...



The screenshot shows the Edge Impulse deployment interface. At the top, there are tabs for "Computer" and "Mobile phone". The URL in the address bar is studio.edgeimpulse.com/studio/114253/deployment. Below the tabs, there's a section titled "Select optimizations (optional)". It includes a note about model optimizations and a toggle switch for "Enable EON™ Compiler". A modal window titled "Built Arduino library" is displayed, containing instructions to add the library via the Arduino IDE's "Sketch > Include Library > Add .ZIP Library..." menu. It also mentions examples under "File > Examples > IESTI01 - classification_inferencing". The "Build output" section shows logs for creating a firmware image and writing it to a device. A file browser at the bottom shows "ei-iesti01---image-classification-arduino-1.0.1.zip".



Model Inference



The image shows two screenshots of the Arduino IDE. The left screenshot displays the 'Examples' section of the 'File' menu, with the 'nano_ble33_sense_camera' example highlighted. The right screenshot shows the code for 'nano_ble33_sense_camera' in the main editor window, followed by a terminal window showing the upload process and execution results.

```
/* Includes */
#include <IESTI01__Image_Classification_inferencing.h>
#include <Arduino_OV767X.h>

/* Constant variables */
#define EI_CAMERA_RAW_FRAME_BUFFER_COLS 160
#define EI_CAMERA_RAW_FRAME_BUFFER_ROWS 120

#define DWORD_ALIGN_PTR(a) ((a & 0x3) ?((uintptr_t)a | 0x4) : (a))

/*
** NOTE: If you run into TFLite arena allocation issue.
** This may be due to memory fragmentation.
** Try defining "-DEI_CLASSIFIER_ALLOCATION_STATIC" in
** if it doesn't exist) and copy this file to
** /lib/tflite/classifiers
*/

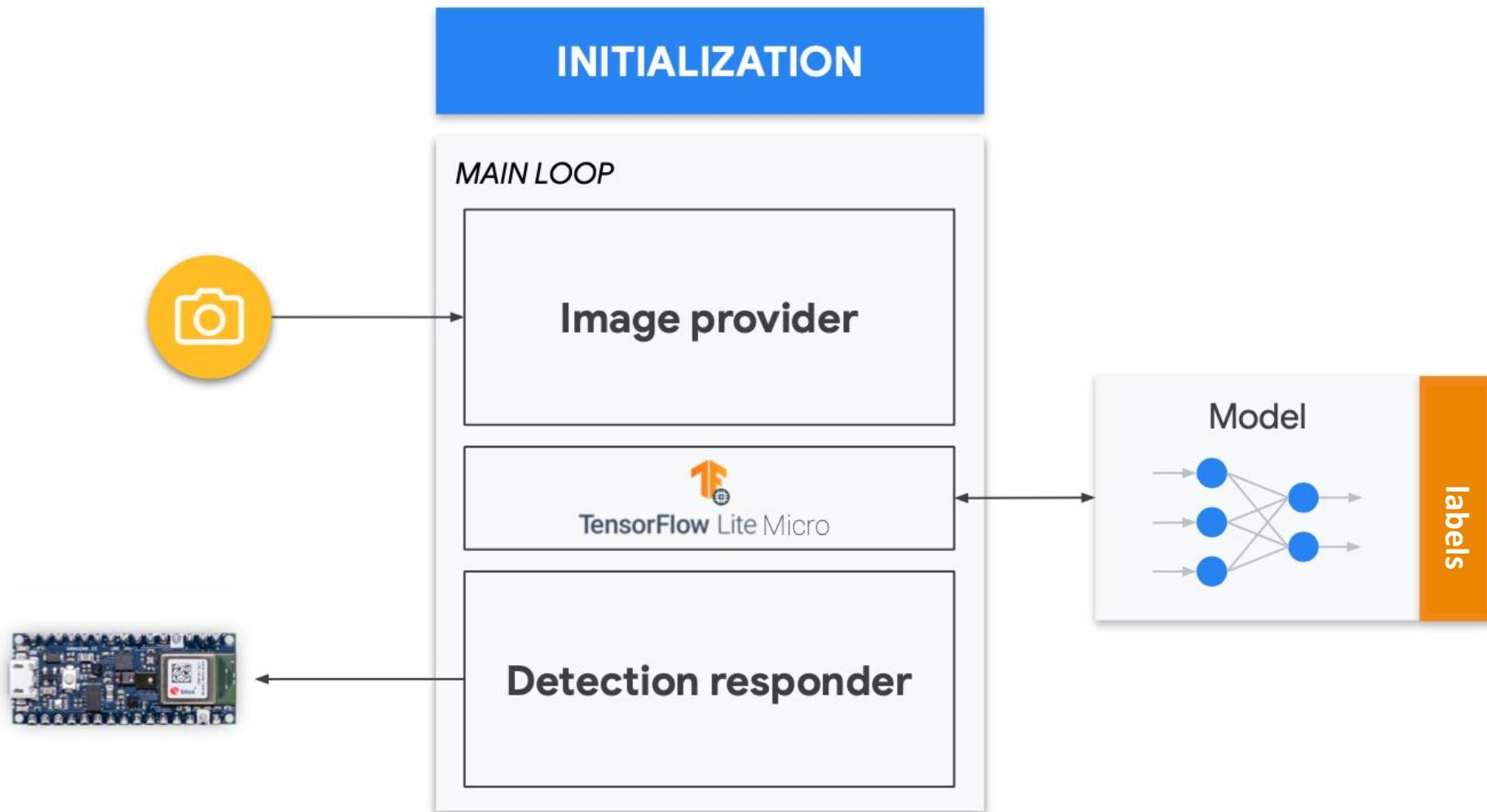
Done uploading.

Done in 0.001 seconds
Write 344544 bytes to flash (85 pages)
[=====] 100% (85/85 pages)
Done in 13.663 seconds
```

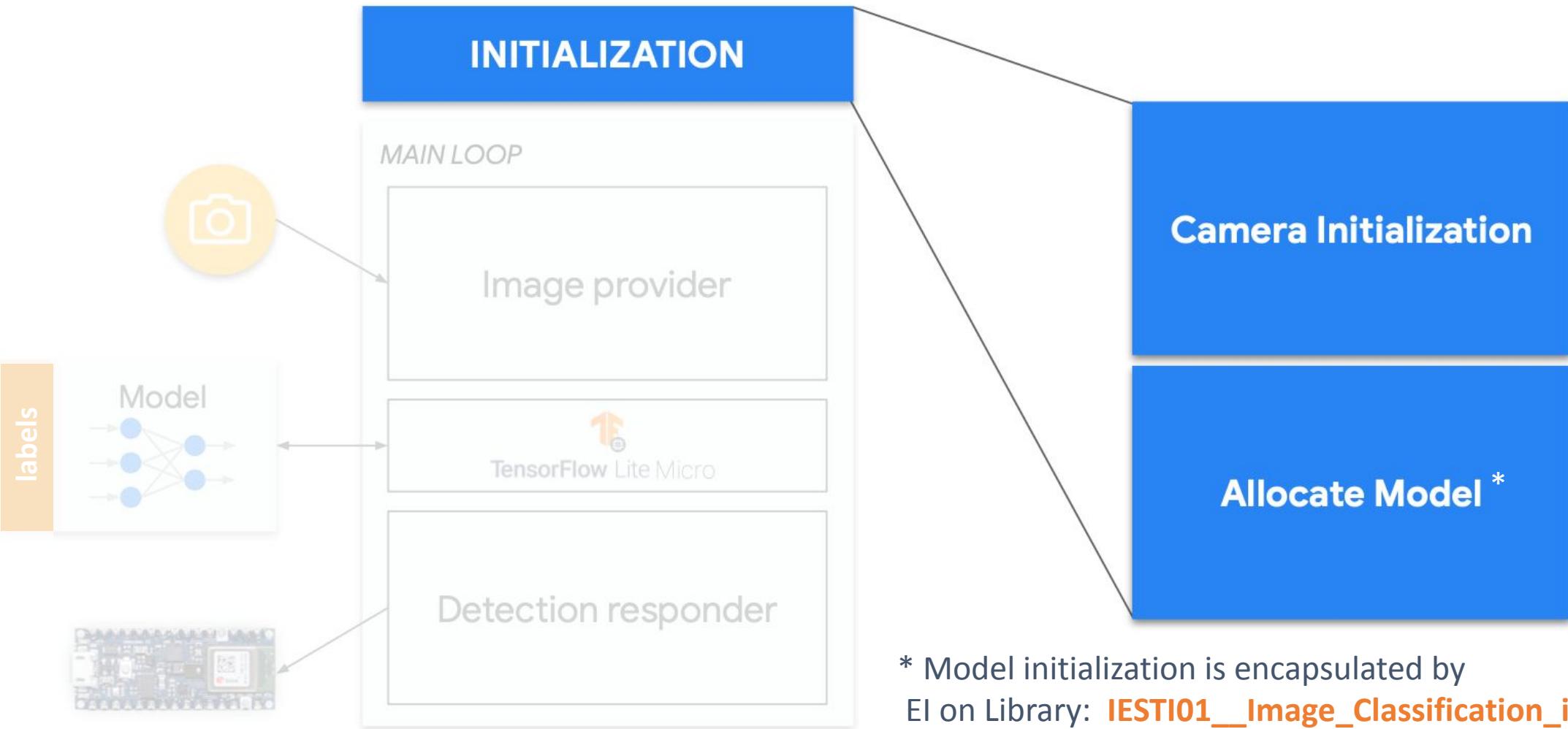
1 Arduino Nano 33 BLE on /dev/cu.usbmodem1201

Img. Classification Components

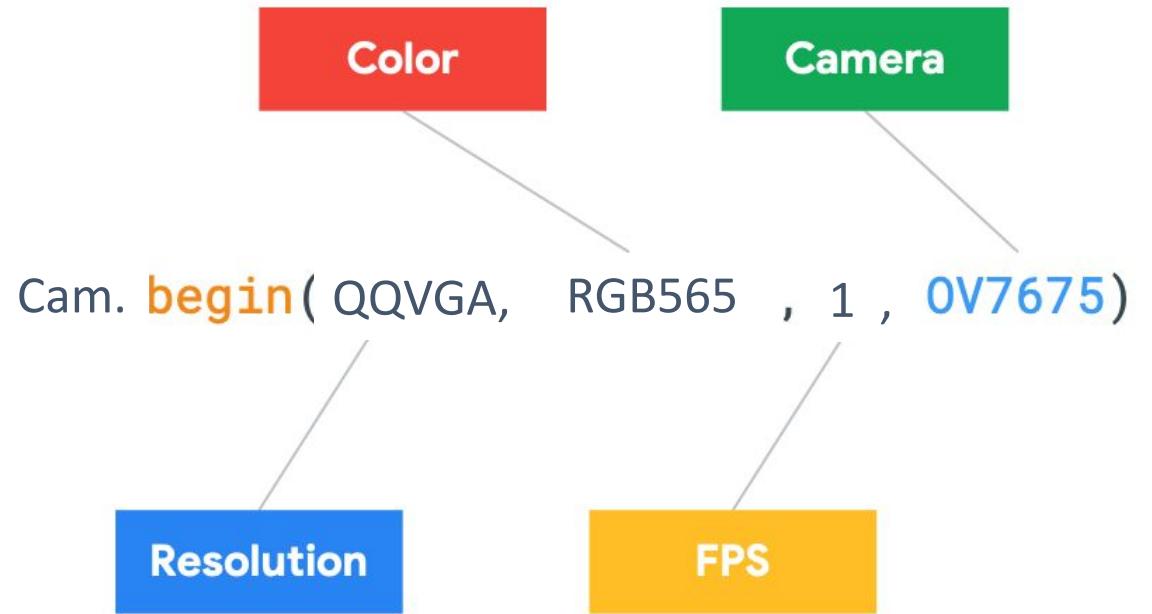
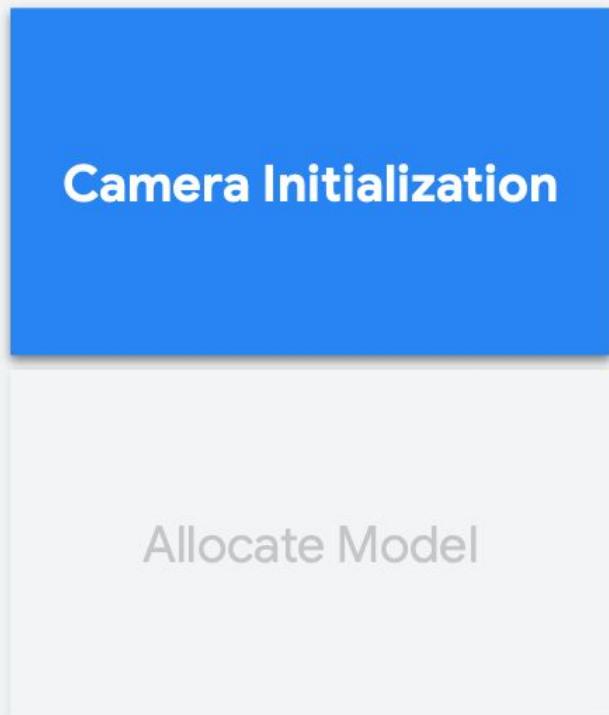
(nano_ble33_sense_camera.ino)



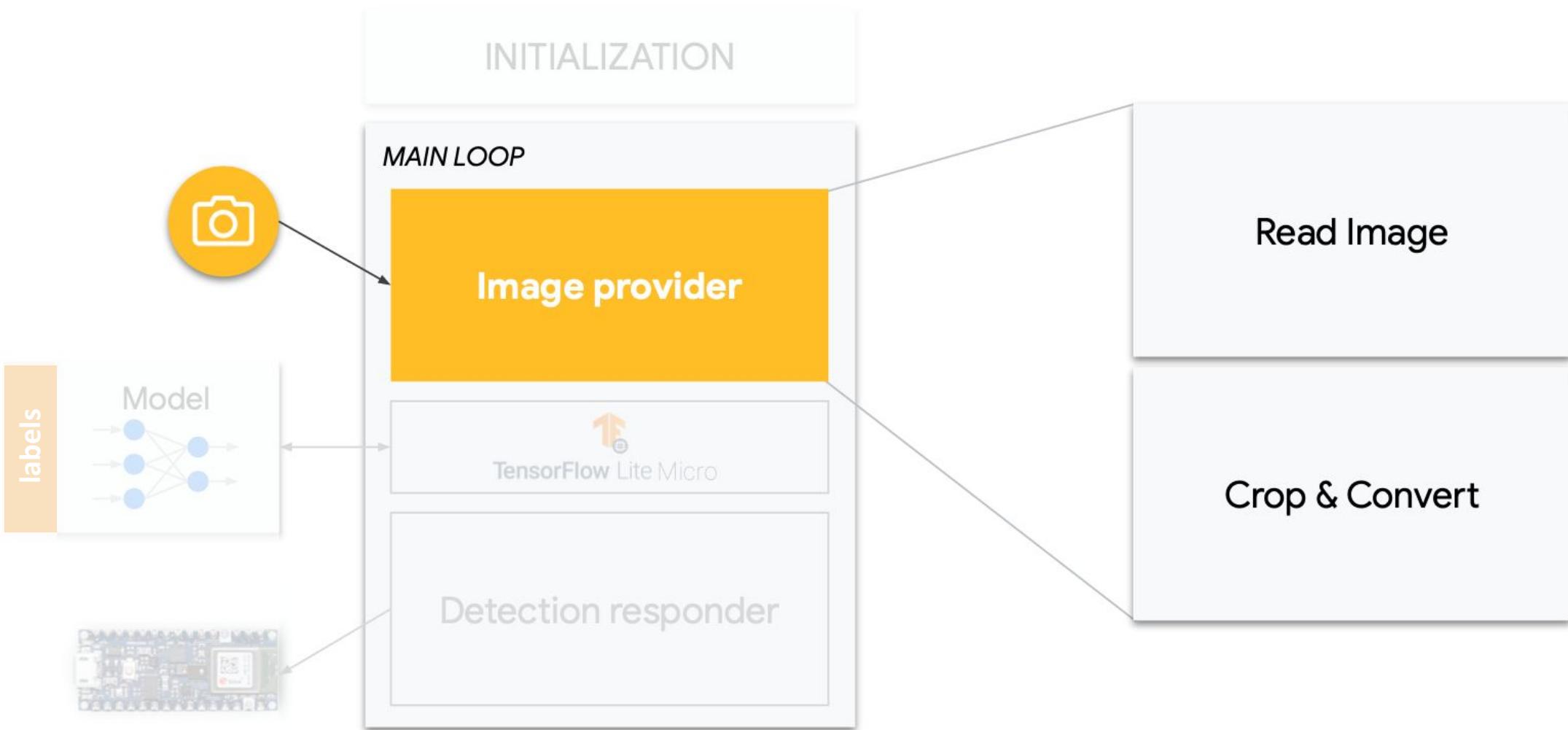
Initialization



Initialization



Pre-processing



Pre-processing

Read Image

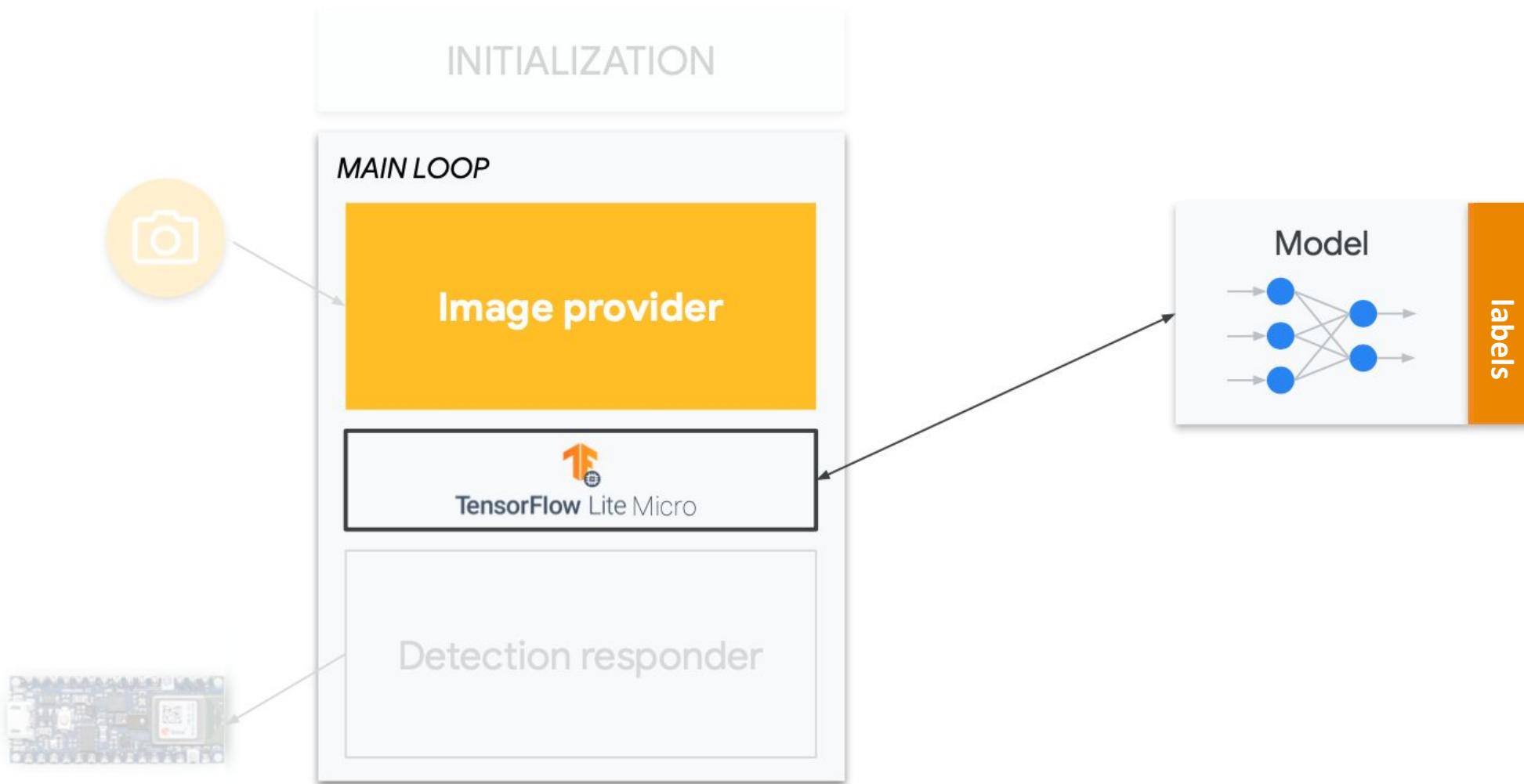
Crop & Convert

```
void *snapshot_mem = NULL;
uint8_t *snapshot_buf = NULL;
snapshot_mem = ei_malloc(resize_col_sz*resize_row_sz*2);
if(snapshot_mem == NULL) {
    ei_printf("failed to create snapshot_mem\r\n");
    break;
}
snapshot_buf = (uint8_t *)DWORD_ALIGN_PTR((uintptr_t)snapshot_mem);

if (ei_camera_capture(EI_CLASSIFIER_INPUT_WIDTH,
    EI_CLASSIFIER_INPUT_HEIGHT, snapshot_buf) == false) {
    ei_printf("Failed to capture image\r\n");
    if (snapshot_mem) ei_free(snapshot_mem);
    break;
}

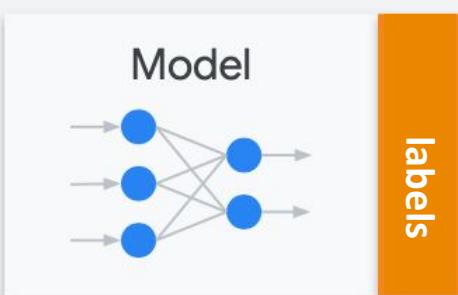
ei::signal_t signal;
signal.total_length = EI_CLASSIFIER_INPUT_WIDTH * EI_CLASSIFIER_INPUT_HEIGHT;
signal.get_data = &ei_camera_cutout_get_data;
```

Interpreter + Model



Interpreter + Model

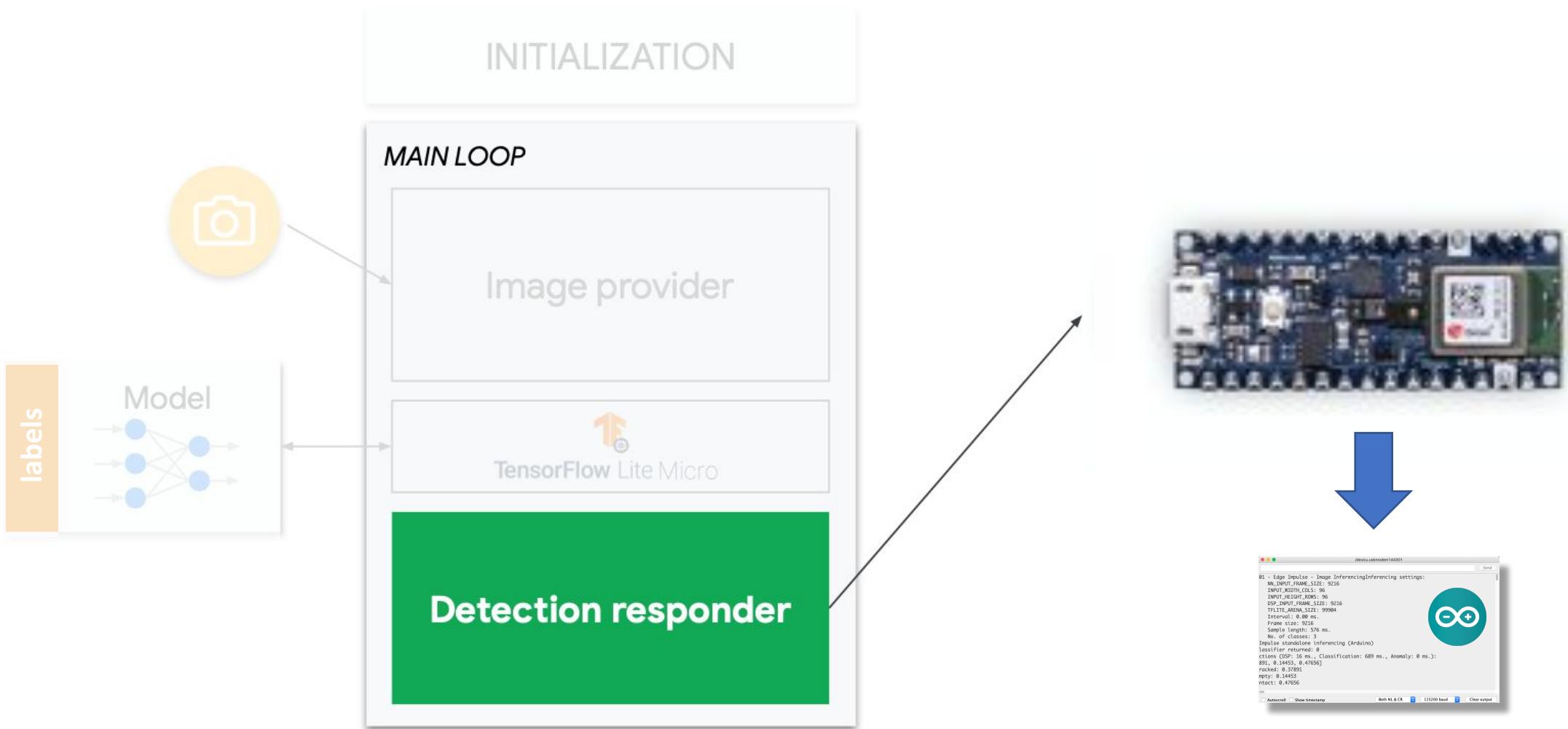

TensorFlow Lite Micro



```
// run the impulse: DSP, neural network and the Anomaly algorithm
ei_impulse_result_t result = { 0 };

EI_IMPULSE_ERROR ei_error = run_classifier(&signal, &result, debug_nn);
if (ei_error != EI_IMPULSE_OK) {
    ei_printf("Failed to run impulse (%d)\n", ei_error);
    ei_free(snapshot_mem);
    break;
}
```

Post-processing



Detection responder

```
// print the predictions
ei_printf("Predictions (DSP: %d ms., Classification: %d ms., Anomaly: %d ms.): \n",
          result.timing.dsp, result.timing.classification, result.timing.anomaly);
#if EI_CLASSIFIER_OBJECT_DETECTION == 1
    bool bb_found = result.bounding_boxes[0].value > 0;
    for (size_t ix = 0; ix < EI_CLASSIFIER_OBJECT_DETECTION_COUNT; ix++) {
        auto bb = result.bounding_boxes[ix];
        if (bb.value == 0) {
            continue;
        }

        ei_printf("    %s (%f) [ x: %u, y: %u, width: %u, height: %u ]\n",
                  bb.label, bb.value, bb.x, bb.y, bb.width, bb.height);
    }

    if (!bb_found) {
        ei_printf("    No objects found\n");
    }
#else
    for (size_t ix = 0; ix < EI_CLASSIFIER_LABEL_COUNT; ix++) {
        ei_printf("    %s: %.5f\n", result.classification[ix].label,
                  result.classification[ix].value);
    }

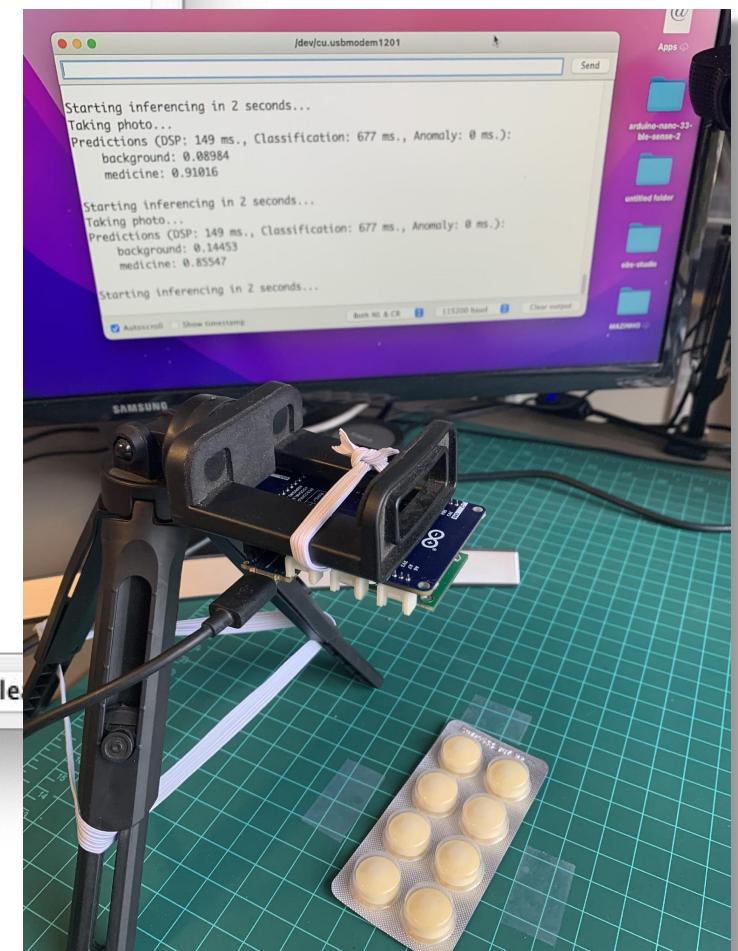
```

```
/dev/cu.usbmodem1201
Starting inferencing in 2 seconds...
Taking photo...
Predictions (DSP: 149 ms., Classification: 677 ms., Anomaly: 0 ms.):
background: 0.10938
medicine: 0.89062

Starting inferencing in 2 seconds...
Taking photo...
Predictions (DSP: 149 ms., Classification: 677 ms., Anomaly: 0 ms.):
background: 0.10547
medicine: 0.89453

Starting inferencing in 2 seconds...
Taking photo...

 Autoscroll  Show timestamp
Both NL & CR
115200 baud
Clear
```



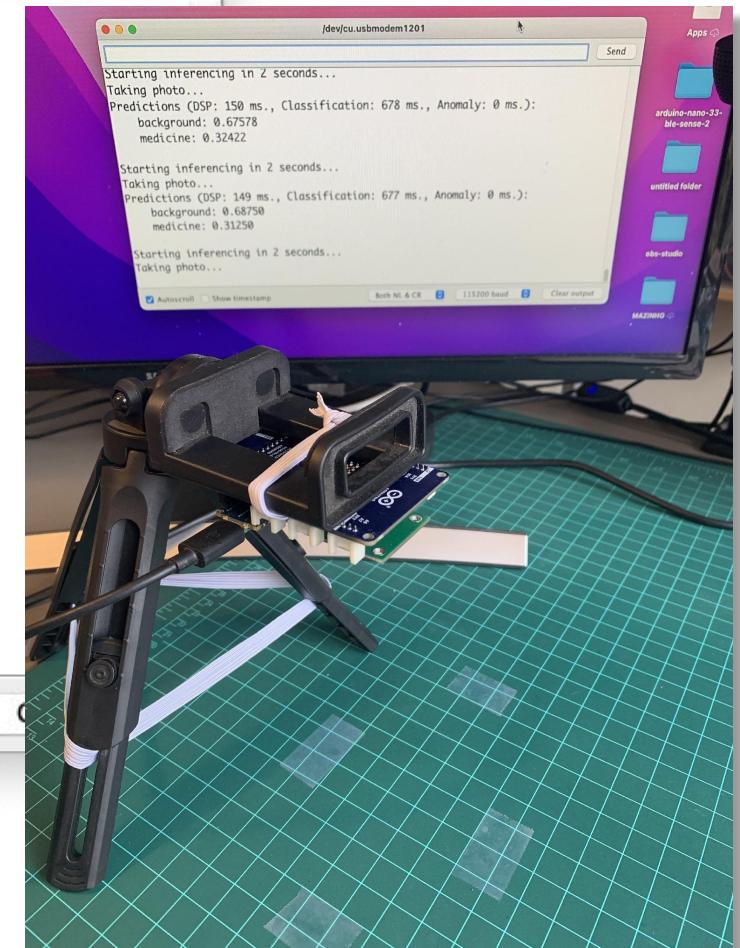
```
/dev/cu.usbmodem1201
Send

Starting inferencing in 2 seconds...
Taking photo...
Predictions (DSP: 149 ms., Classification: 678 ms., Anomaly: 0 ms.):
background: 0.69922
medicine: 0.30078

Starting inferencing in 2 seconds...
Taking photo...
Predictions (DSP: 149 ms., Classification: 678 ms., Anomaly: 0 ms.):
background: 0.71484
medicine: 0.28516

Starting inferencing in 2 seconds...

 Autoscroll  Show timestamp Both NL & CR 115200 baud
```



Thanks



UNIFEI

