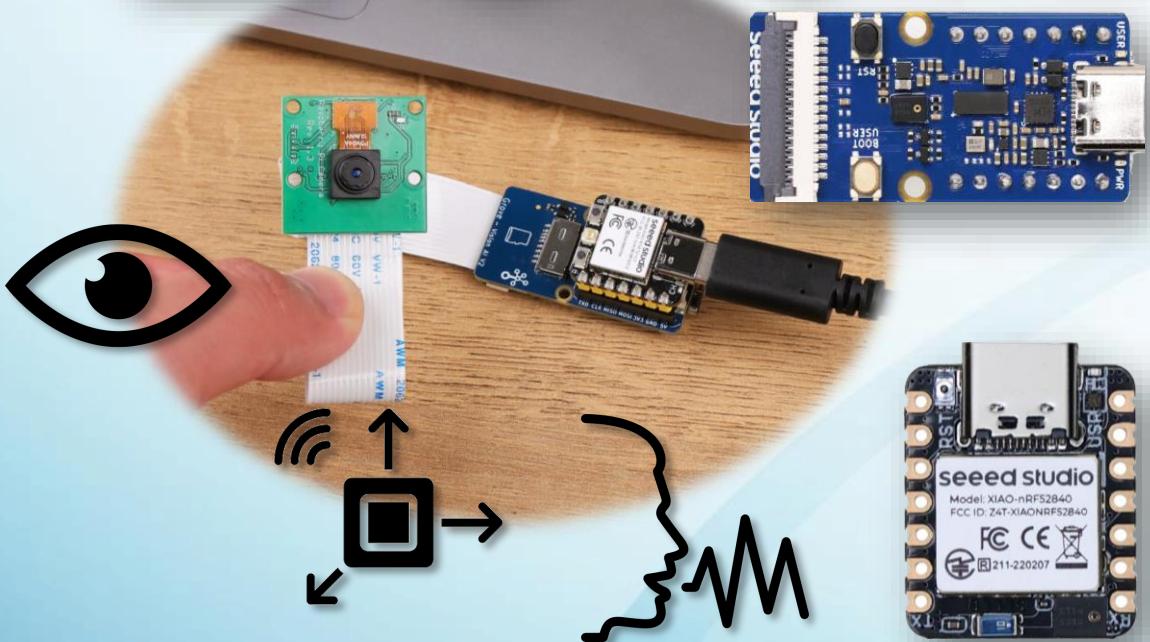


OmniXRI TinyML 小學堂 2025



Himax



歡迎加入
邊緣人俱樂部



沒有最邊



只有更邊

Cortex-M
Processor



Ethos-U
MicroNPU

【第 6 講】
TinyML 開發流程



歐尼克斯實境互動工作室 (OmniXRI Studio)
許哲豪 (Jack Hsu)

簡報大綱



- 6.1. Edge Impulse簡介
- 6.2. 資料集建立
- 6.3. 模型選用與訓練
- 6.4. 模型優化及部署

本課程完全免費，請勿移作商業用途！
歡迎留言、訂閱、點讚、轉發，讓更多需要的朋友也能一起學習。

完整課程大綱：<https://omnixri.blogspot.com/2025/03/omnixri-tinyml-2025-0.html>
課程直播清單：<https://www.youtube.com/@omnixri1784streams>



6.1. Edge Impulse 簡介

TinyML 開發流程 (簡易版)



Edge Impulse Studio

MCU等級AI供應商為tinyML主要成員

提供眾多API及SDK方便開發應用

- 動作辨識、關鍵詞偵測、物件偵測等
- 支援多種小型AI及MCU / MPU開發
- 提供豐富線上開發工具

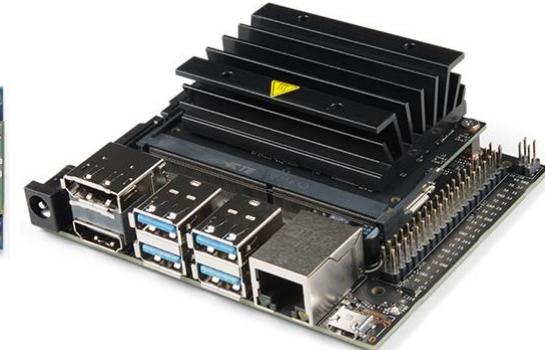


**Seeed Grove Vision
AI Module V2 kit**

資料來源：<https://docs.edgeimpulse.com/docs/>



**Arduino Nano
33 BLE Sense**



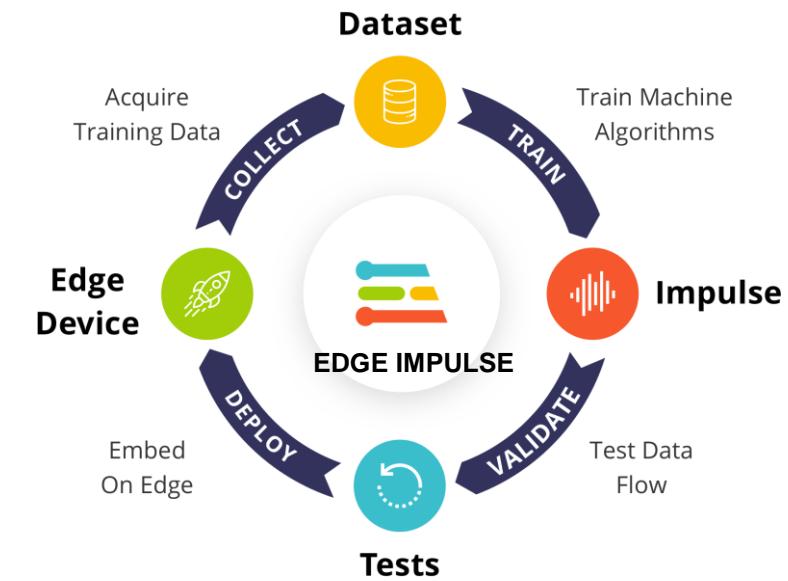
Nvidia Jetson Nano



Raspberry Pi 4



**Seeed Xiao
nRF52840 Sense**



Edge Impulse - 主要功能

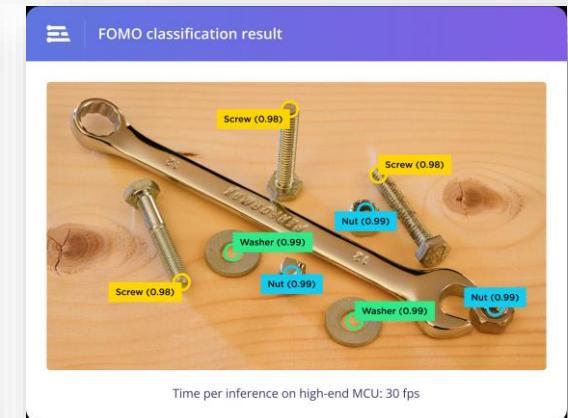
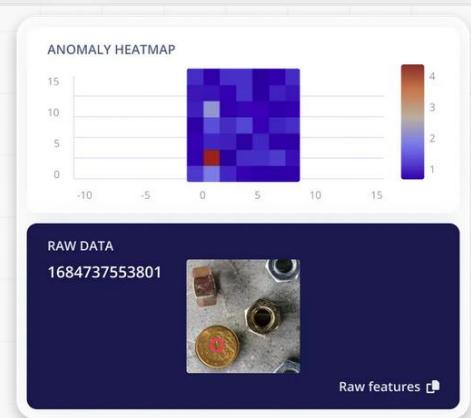
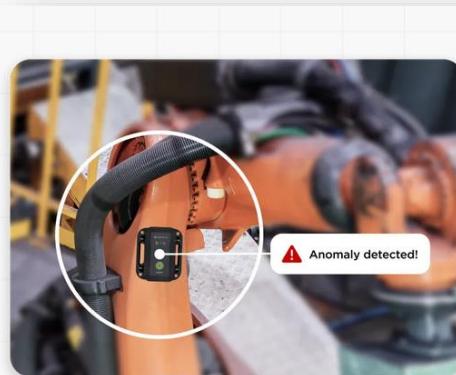
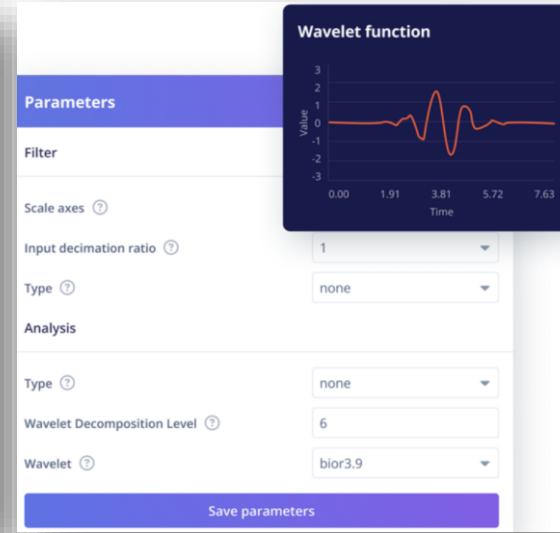
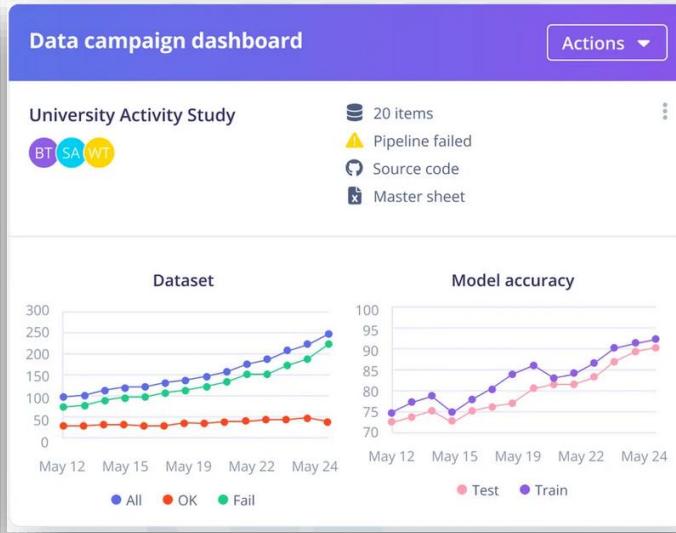
```
# Quick integration for any Python script
import edgeimpulse as ei

# 1. Set API key from Edge Impulse project
ei.API_KEY = "ei_dae2..."

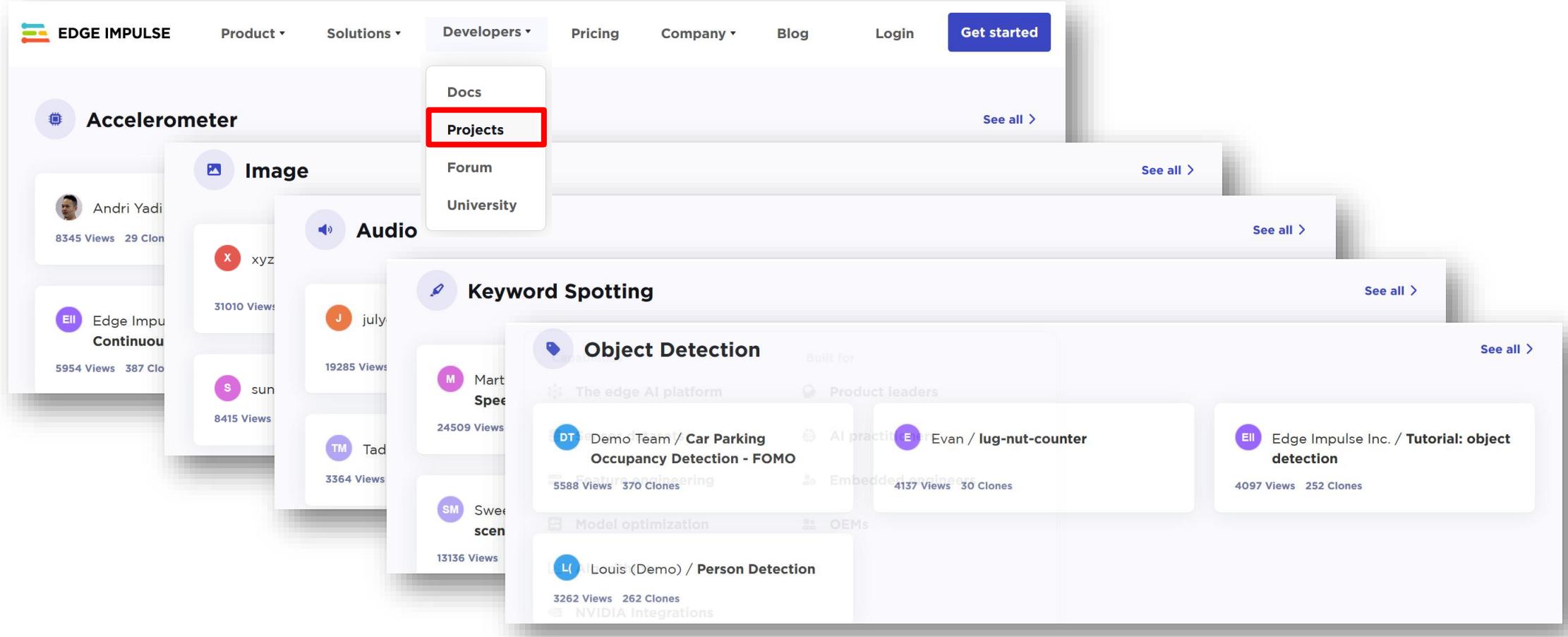
# 2. Profile model performance on target device
results = ei.model.profile(model="/path/to/model",
                           device="cortex-m4f-80mhz")

# 3. View profiling results
results.summary()

# 4. Optimize model and convert to C++ library
from ei.model.output_type import Classification
ei.model.deploy(model="/path/to/model",
               model_output_type=Classification(),
               output_directory="/library")
```



Edge Impulse – 公用專案

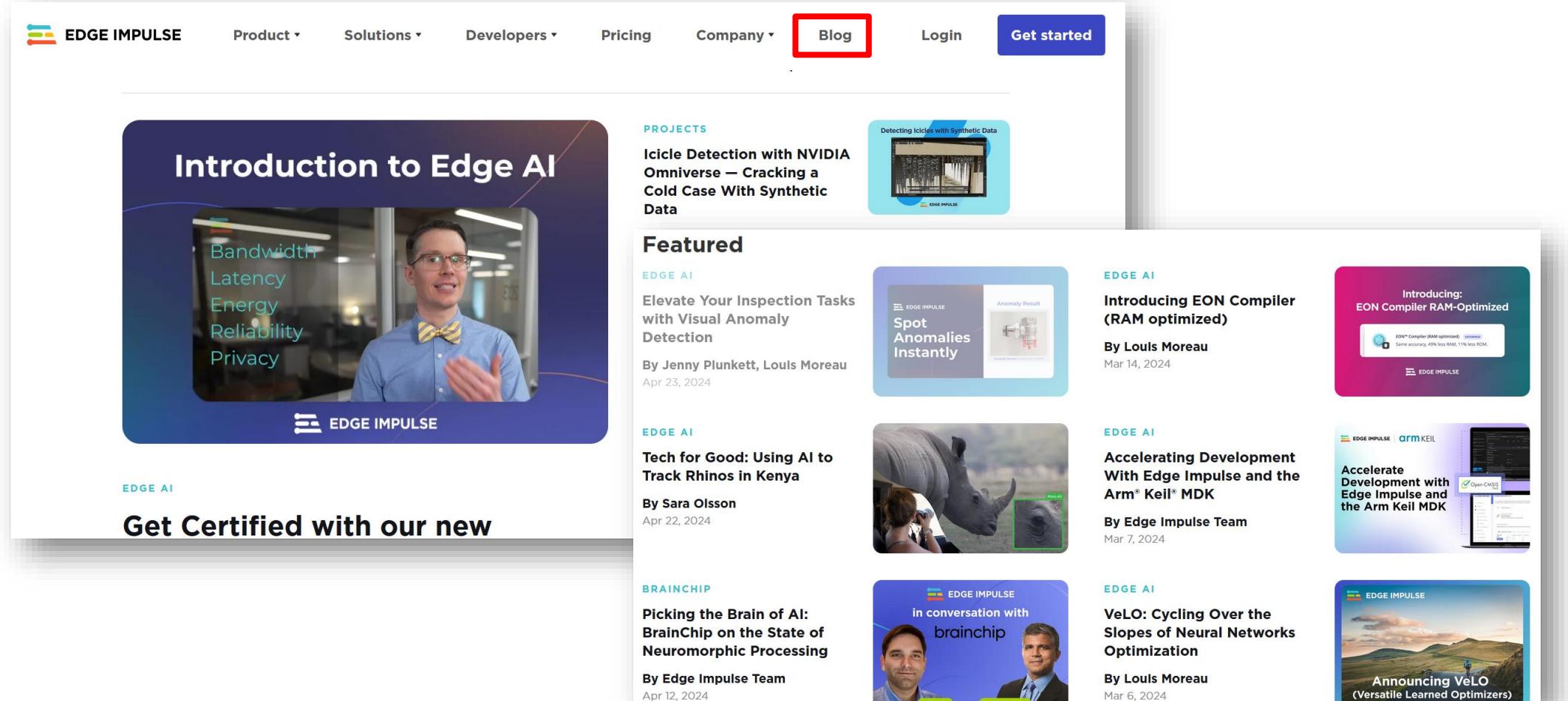


The screenshot shows the Edge Impulse website's main navigation bar with links for Product, Solutions, Developers (highlighted), Pricing, Company, Blog, Login, and Get started. A dropdown menu for 'Developers' is open, showing options like Docs, Projects (which is highlighted with a red box), Forum, and University. Below this, there are several project cards categorized by type: Accelerometer, Image, Audio, Keyword Spotting, Object Detection, and others. Each card displays the project name, author, views, and clones.

Category	Project Name	Author	Views	Clones
Accelerometer	Andri Yadi		8345	29
	Edge Impulse Continuous		5954	387
Image	xyz		31010	
	July	Mart Spec	19285	
Audio	sun	Sweet scen	8415	
	Tad	Louis (Demo)	3364	
Keyword Spotting		Demo Team / Car Parking Occupancy Detection - FOMO	5588	370
		Evan / lug-nut-counter	4137	30
Object Detection		Feature engineering	3262	262
		NVIDIA Integrations		

<https://edgeimpulse.com/projects/overview>

Edge Impulse – 部落格文章



The screenshot shows the Edge Impulse blog page. At the top, there is a navigation bar with links: EDGE IMPULSE, Product, Solutions, Developers, Pricing, Company, **Blog** (which is highlighted with a red box), Login, and Get started.

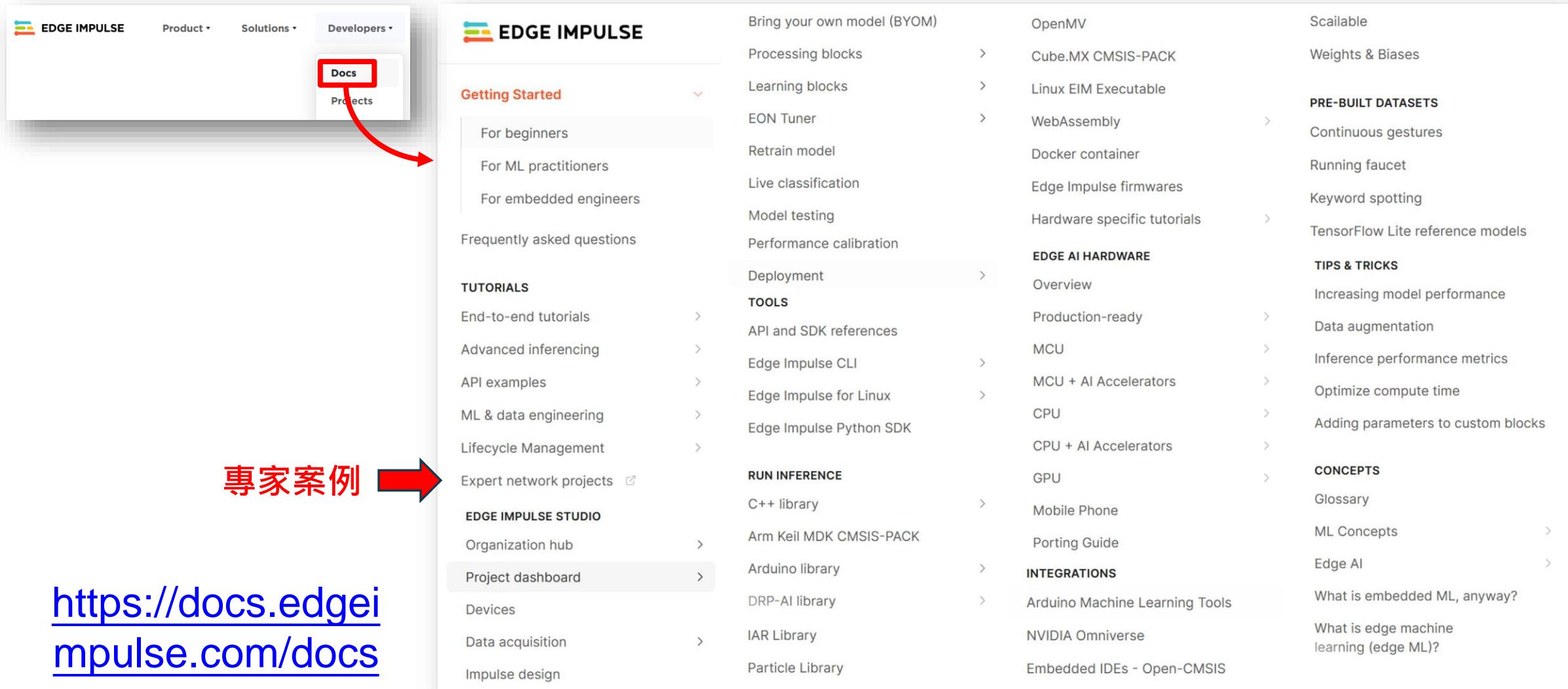
The main content area features a large banner for "Introduction to Edge AI" with a video thumbnail of a man speaking. Below the banner, there is a section titled "PROJECTS" with a thumbnail for "Detecting Icicles with Synthetic Data".

The "Featured" section contains several blog posts:

- EDGE AI**: "Elevate Your Inspection Tasks with Visual Anomaly Detection" by Jenny Plunkett, Louis Moreau (Apr 23, 2024)
- EDGE AI**: "Tech for Good: Using AI to Track Rhinos in Kenya" by Sara Olsson (Apr 22, 2024)
- BRAINCHIP**: "Picking the Brain of AI: BrainChip on the State of Neuromorphic Processing" by Edge Impulse Team (Apr 12, 2024)
- EDGE AI**: "Introducing EON Compiler (RAM optimized)" by Louis Moreau (Mar 14, 2024)
- EDGE AI**: "Accelerating Development With Edge Impulse and the Arm® Keil® MDK" by Edge Impulse Team (Mar 7, 2024)
- EDGE AI**: "VeLO: Cycling Over the Slopes of Neural Networks Optimization" by Louis Moreau (Mar 6, 2024)
- EDGE AI**: "Announcing VeLO (Versatile Learned Optimizers)" (Thumbnail only, no title or author information visible)

<https://edgeimpulse.com/blog/>

Edge Impulse – 說明文件



<https://docs.edgeimpulse.com/docs>

EDGE IMPULSE Product ▾ Solutions ▾ Developers ▾

EDGE IMPULSE

Getting Started

- For beginners
- For ML practitioners
- For embedded engineers
- Frequently asked questions

TUTORIALS

- End-to-end tutorials
- Advanced inferencing
- API examples
- ML & data engineering
- Lifecycle Management
- Expert network projects

EDGE IMPULSE STUDIO

- Organization hub
- Project dashboard
- Devices
- Data acquisition
- Impulse design

Bring your own model (BYOM)

- Processing blocks
- Learning blocks
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Performance calibration
- Deployment
- API and SDK references
- Edge Impulse CLI
- Edge Impulse for Linux
- Edge Impulse Python SDK
- C++ library
- Arm Keil MDK CMSIS-PACK
- Arduino library
- DRP-AI library
- IAR Library
- Particle Library

OpenMV

- Cube.MX CMSIS-PACK
- Linux EIM Executable
- WebAssembly
- Docker container
- Edge Impulse firmwares
- Hardware specific tutorials
- Overview
- Production-ready
- MCU
- MCU + AI Accelerators
- CPU
- CPU + AI Accelerators
- GPU
- Mobile Phone
- Porting Guide
- Arduino Machine Learning Tools
- NVIDIA Omniverse
- Embedded IDEs - Open-CMSIS

Scalable

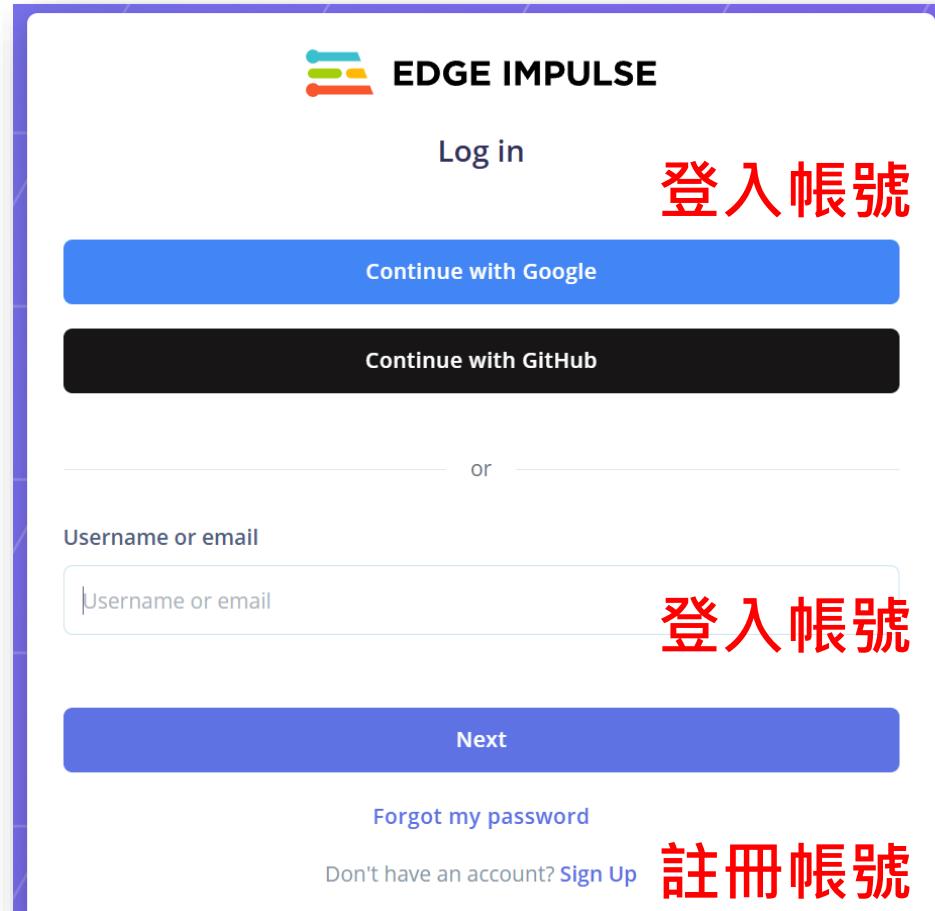
- Weights & Biases
- PRE-BUILT DATASETS
- Continuous gestures
- Running faucet
- Keyword spotting
- TensorFlow Lite reference models
- TIPS & TRICKS
- Increasing model performance
- Data augmentation
- Inference performance metrics
- Optimize compute time
- Adding parameters to custom blocks
- CONCEPTS
- Glossary
- ML Concepts
- Edge AI
- What is embedded ML, anyway?
- What is edge machine learning (edge ML)?

Edge Impulse – Expert network projects

- Featured Machine Learning Projects
- Prototype and Concept Projects
- Image Projects
- Audio Projects
- Predictive Maintenance & Fault Classification
- Accelerometer & Activity Projects
- Air Quality & Environmental Projects
- Novel Sensor Projects
- Software Integration Demos

<https://docs.edgeimpulse.com/experts>

Edge Impulse - 註冊/登入帳號



<https://studio.edgeimpulse.com/login>

免費帳號使用限制

- **2組私人專案**
- 公用專案不限
- 最多**3個成員(限公用專案)**
- 每個專案 CPU 計算用記憶體 **4GB** 或 **4小時**資料
- 每個專案計算時間**20分鐘**
- EON Tuner, Compiler 有限制

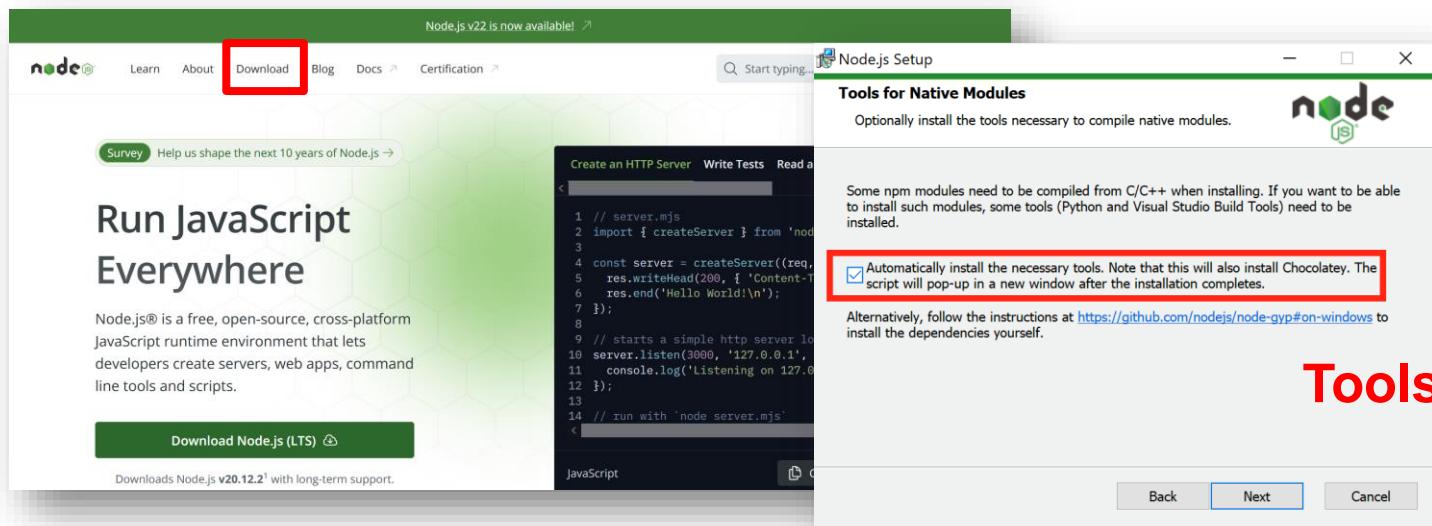
<https://edgeimpulse.com/pricing>

下載及安裝必要工具

1. Python 3.x



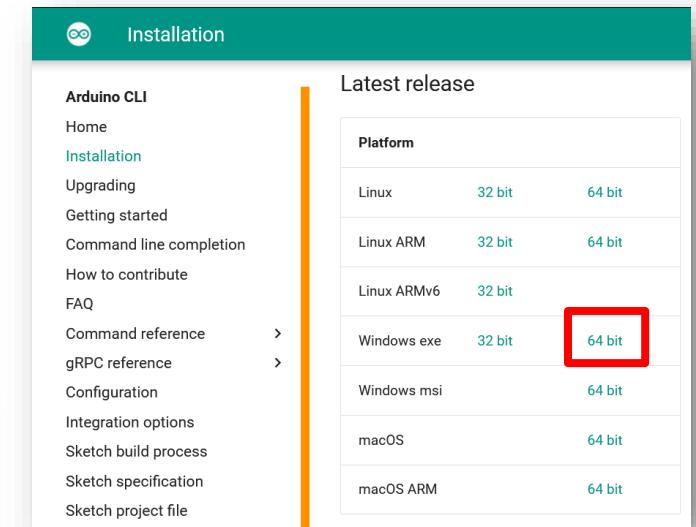
2. Node.js



(若後續 Edge Impulse 無法安裝
可換回舊版 · 例：v18.20.2 LTS)

3. Arduino CLI

(選配 · 依不同開發板需求)



安裝時要注意
記得一定要勾選
Tools for Native Modules
以 node -v 及 npm -v
檢查安裝好的版本

安裝 Edge Impulse 工作環境

➤ 安裝Edge Impulse CLI

```
npm install -g edge-impulse-cli –force
```

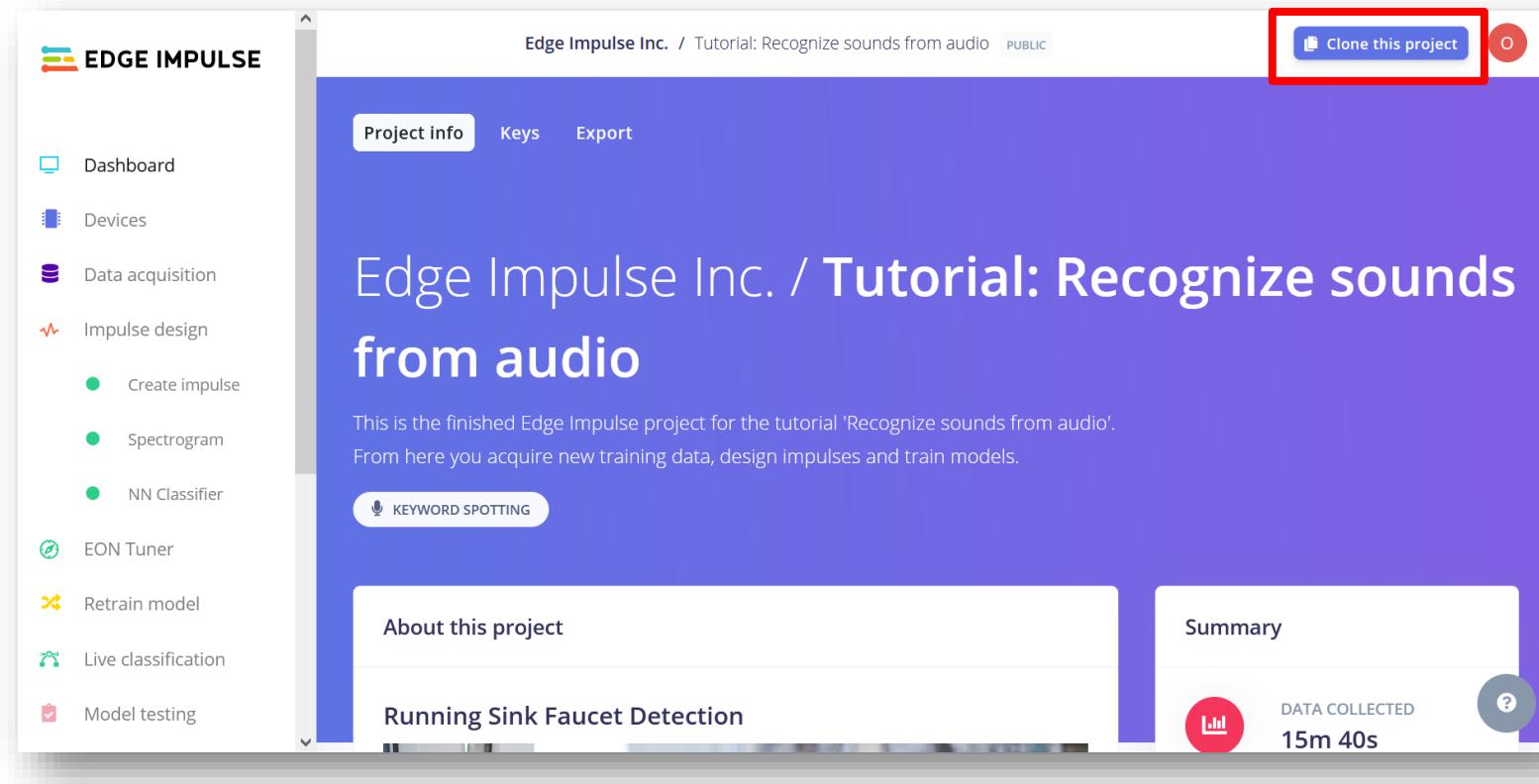
➤ 安裝完成後會得到下列工具

- **edge-impulse-daemon** 透過 COM 埠和雲端平台連接
- **edge-impulse-uploader** 允許上傳本機檔案
- **edge-impulse-data-forwarder** 透過 COM 埠上傳開發板感測器數值
- **edge-impulse-run-impulse** 顯示裝置上運行的模型
- **edge-impulse-blocks** 建立自定義區塊
- **himax-flash-tool** 燒錄 Haimax WE-I Plus

<https://docs.edgeimpulse.com/docs/tools/edge-impulse-cli/cli-installation>

啟動公用專案範例 — 水流聲辨識

Edge Impulse Inc. / Tutorial: Recognize sounds from audio



The screenshot shows the Edge Impulse Studio interface. On the left is a sidebar with various project management options like Dashboard, Devices, Data acquisition, and Model testing. The main area displays a project titled "Tutorial: Recognize sounds from audio". The title is prominently displayed at the top. Below it is a brief description: "This is the finished Edge impulse project for the tutorial 'Recognize sounds from audio'. From here you acquire new training data, design impulses and train models." There are two buttons: "KEYWORD SPOTTING" and "About this project". At the bottom, there's a progress bar labeled "Running Sink Faucet Detection" and a summary box stating "DATA COLLECTED 15m 40s".

複製專案到
個人帳號

水流聲辨識



<https://studio.edgeimpulse.com/public/14301/latest>

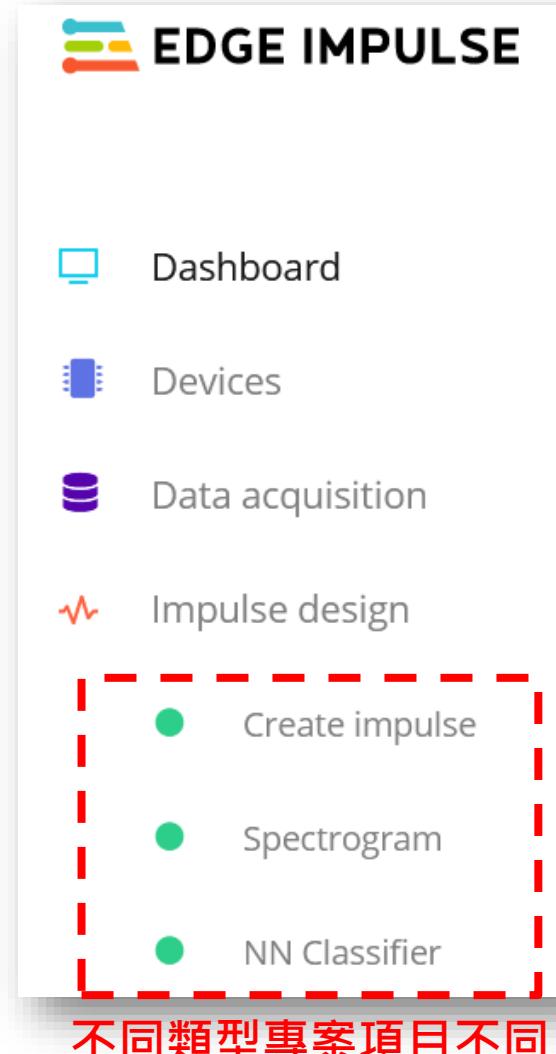
專案工作流程



水流聲辨識

1. 準備開發板
2. 收集資料
3. 建立資料集
4. 設計及選用模型
5. 配置MFE模塊
6. 配置神經網路
7. 分類新資料
8. 模型測試
9. 模型故障排除
10. 部署到指定裝置

Edge Impulse Studio — 專案功能表



專案儀錶板
連線裝置
資料擷取
流程設計
建立流程
資料頻譜
神經網路分類器

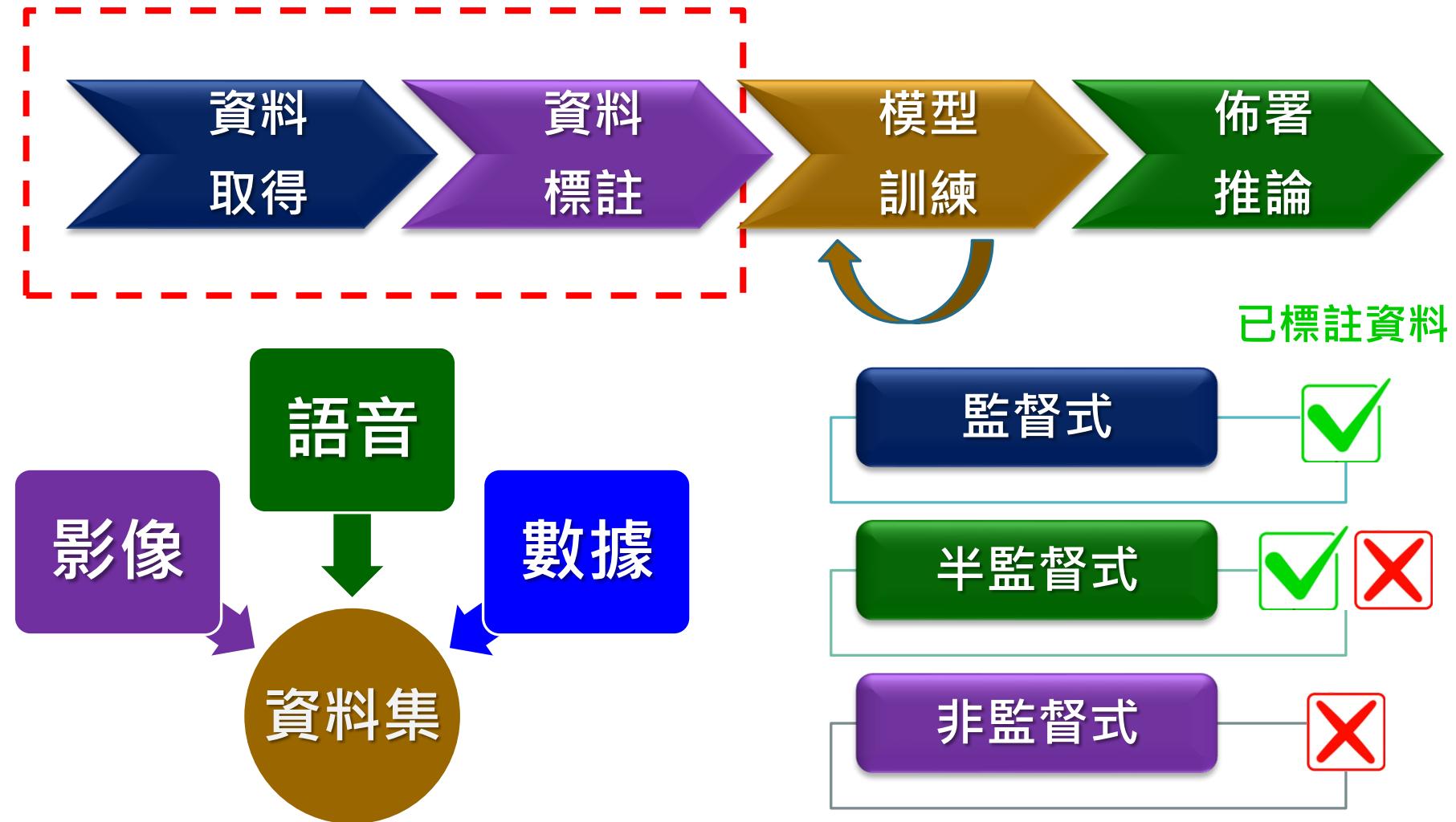
- EON Tuner
- Retrain model
- Live classification
- Model testing
- Performance calibration
- Versioning
- Deployment

EON調整器
重新訓練模型
立即分類
模型測試
效能效正
版本管理
程式部署



6.1. 資料集建立

常見資料類型



常見資料結構

格式化資料

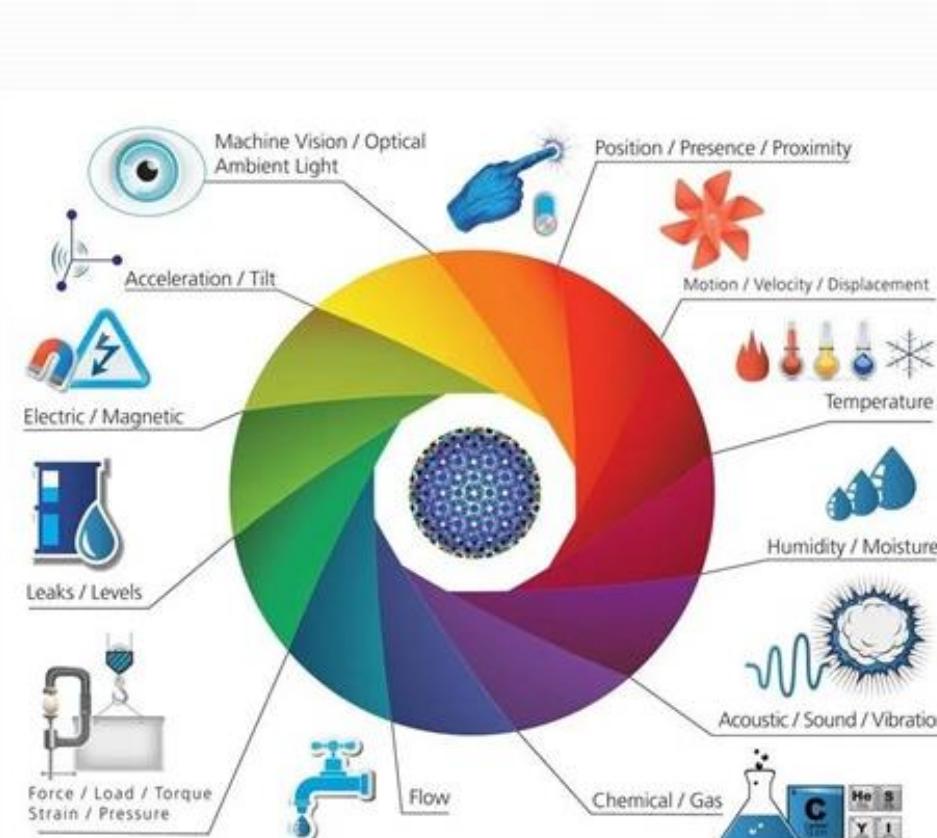
- 一維時序資料（感測器）
光電、溫度、濕度、壓力、
聲音、速度、運動...
- 二維 / 三維影像資料
可見光、紅外光、深度影像
- 影像序列資料（影片）
原始、壓縮、串流
- 多維表單數據（資料庫）

非格式化資料

- 自然語言、文章
- 影像、影片內容
- 非定時定量收集的資料
- 偶發性資料



常見感測器



影像來源：<https://www.edntaiwan.com/20190201ta31-designers-guide-to-iiot-sensor-systems/>

資料來源：<https://omnixri.blogspot.com/2022/11/20221108.html>

- A. 電阻、電容、電感式
- B. 光電式
 - 發射接收式、紅外熱幅射式
- C. 壓電式
- D. 電聲式
 - 超音波、麥克風
- E. 微機電式
 - 運動類、環境類
- F. 電磁波式
- G. 影像式

資料取得方式

資料集來源

- 公開資料集
- 私有資料庫
- 網路收集
- 自行拍攝取像
 - 專業相機（手機、相機、攝影機）
 - 網路攝影機（USB）
 - 開發板專用相機（CSI）
 - ◆ 可見光
 - ◆ 紅外線
- 感測器資料
 - 網路連接、MQTT

取像要領

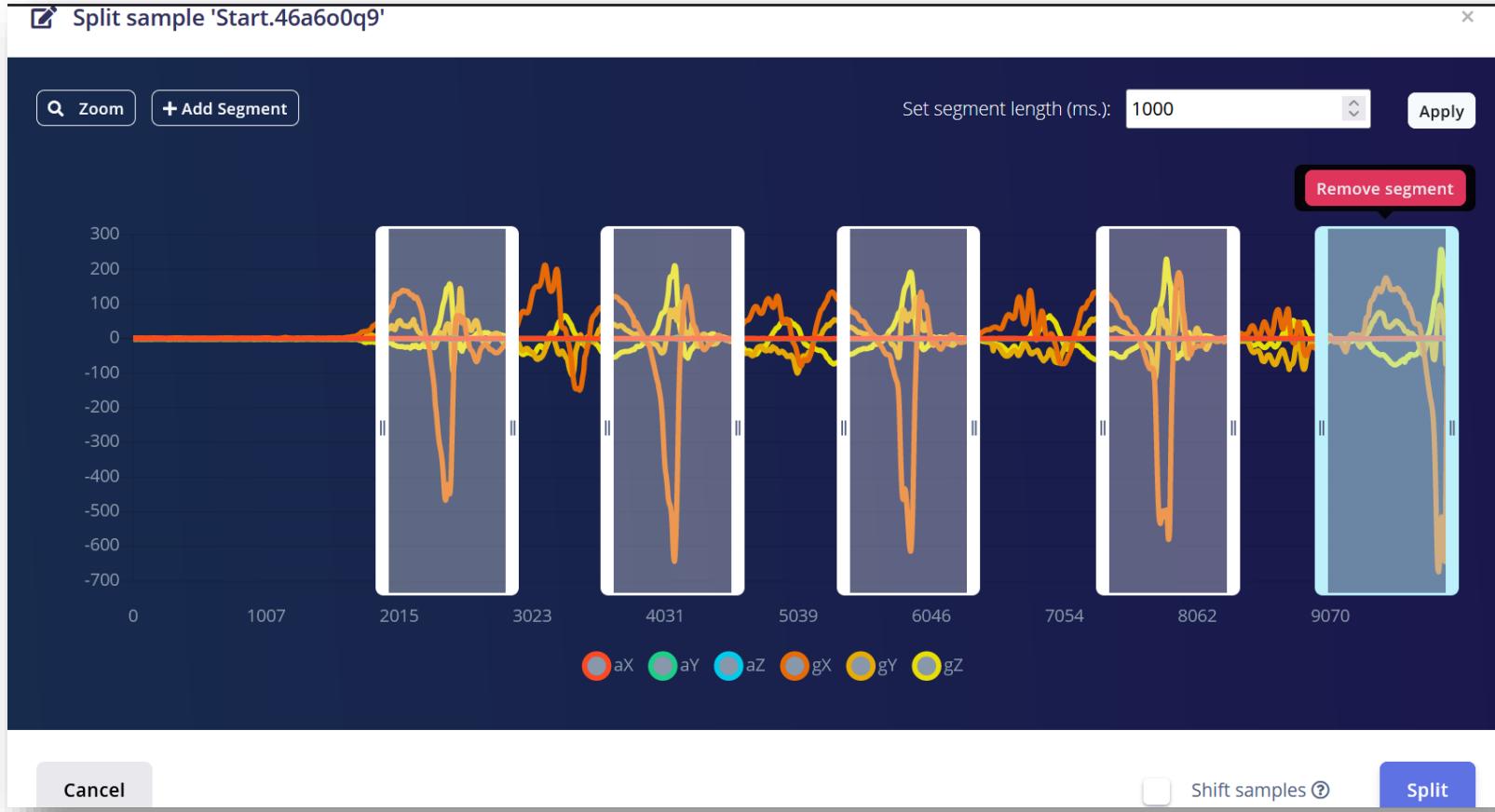
- 取像品質近似
 - 解析度、模糊度...
- 多視點
 - 遠近、角度、物件正反面...
- 多光源差異
 - 環境亮度、陰影、光圈、快門...
- 多背景、品種

**** 建立足夠多樣性的資料集 ****

資料清洗

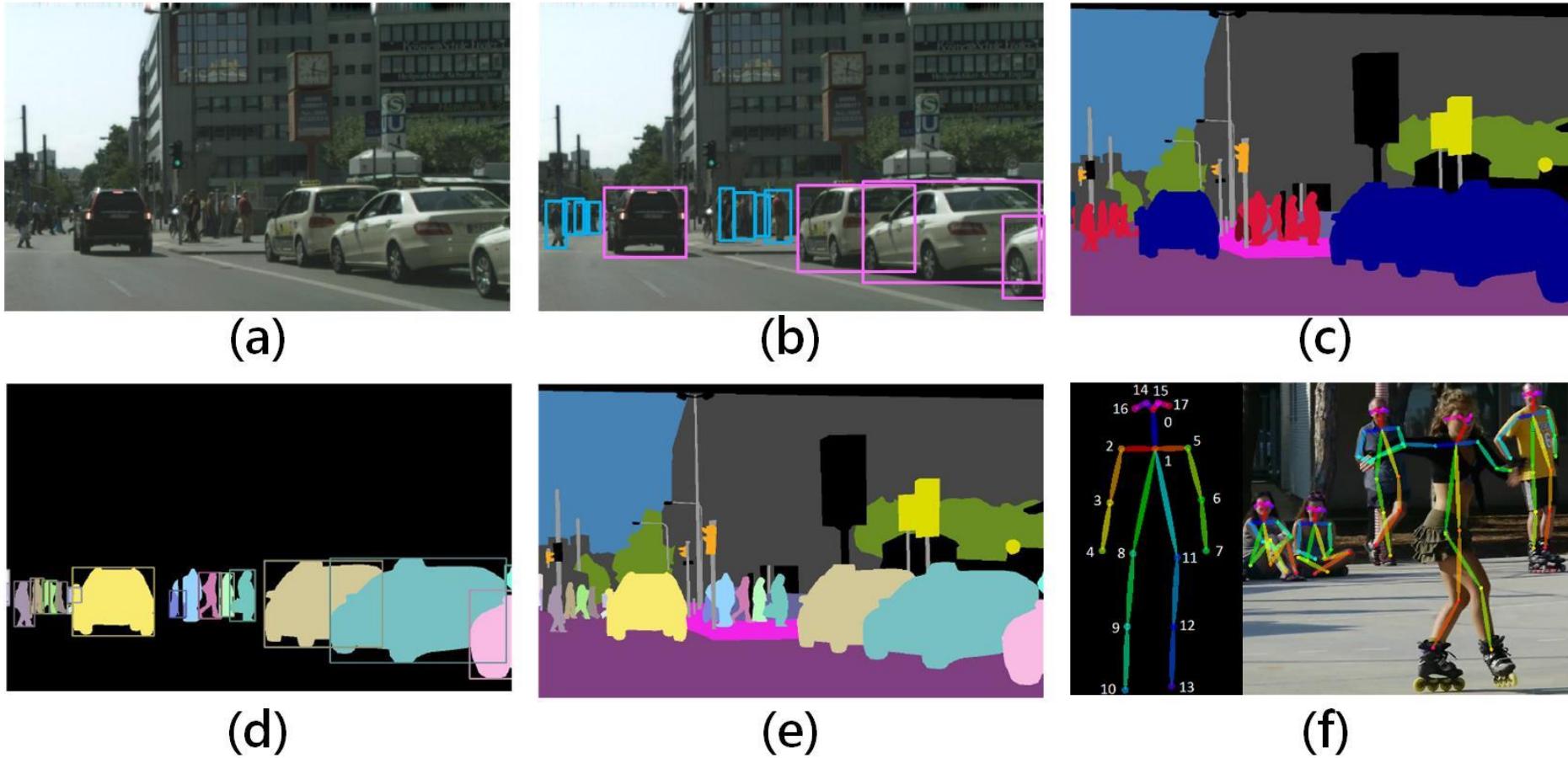
- 去冗、填空、修補、抑雜訊

感測器資料標註



一維？多維？
連續？片段？
手動/自動分割？
特徵時長？

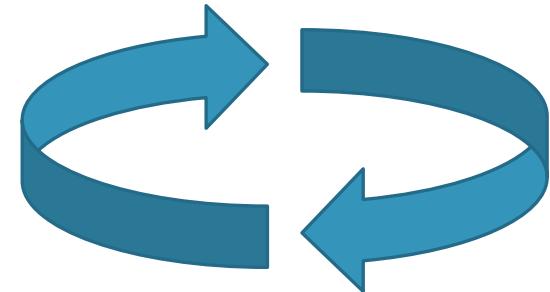
影像資料標註



訓練、驗證及測試資料集

已標註之資料可依一定比例分配到訓練集、驗證集及測試集中。**絕對不可把測試集加入訓練集或驗證集中。**

分配比例**沒有一定標準**，小規模資料集建議**訓練:驗證:測試為6:2:2**。大規模（數百萬筆）資料集，驗證、測試各留數萬筆即可。



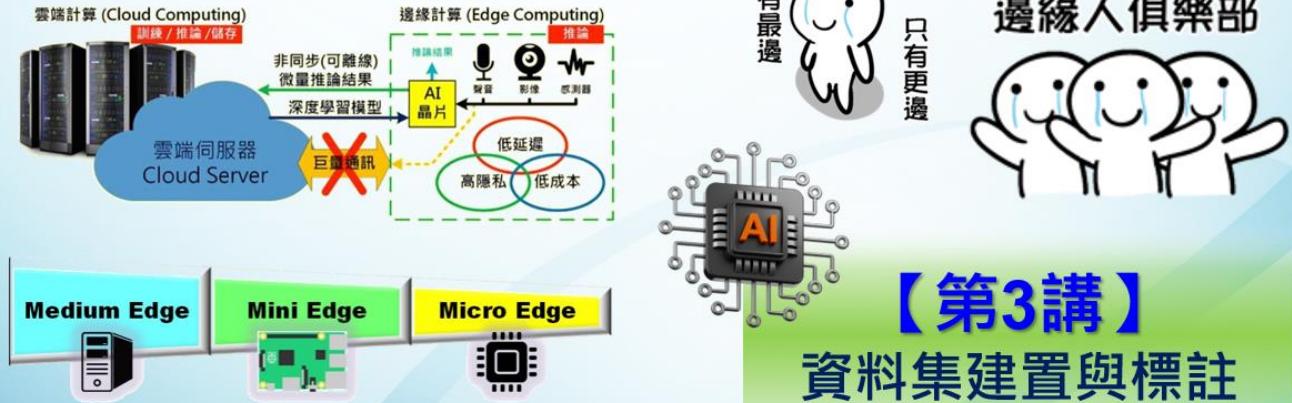
訓練集：就像在學校學習，了解問題，找出學習重點。

驗證集：就像模擬考，測試學習成果，找出修正方向。

測試集：就像大考，出題和模擬考接近年度會影響最後成績。

更多資料集建置與標註

OmniXRI's Edge AI & TinyML 小學堂



歡迎加入
邊緣人俱樂部

【第3講】
資料集建置與標註

歐尼克斯實境互動工作室 (OmniXRI Studio)
許哲豪 (Jack Hsu)

Cloud Computing (Cloud Computing)
Cloud Server

Edge Computing (Edge Computing)
AI 晶片

非同步(可離線)
微量推論結果
深度學習模型

巨量通訊

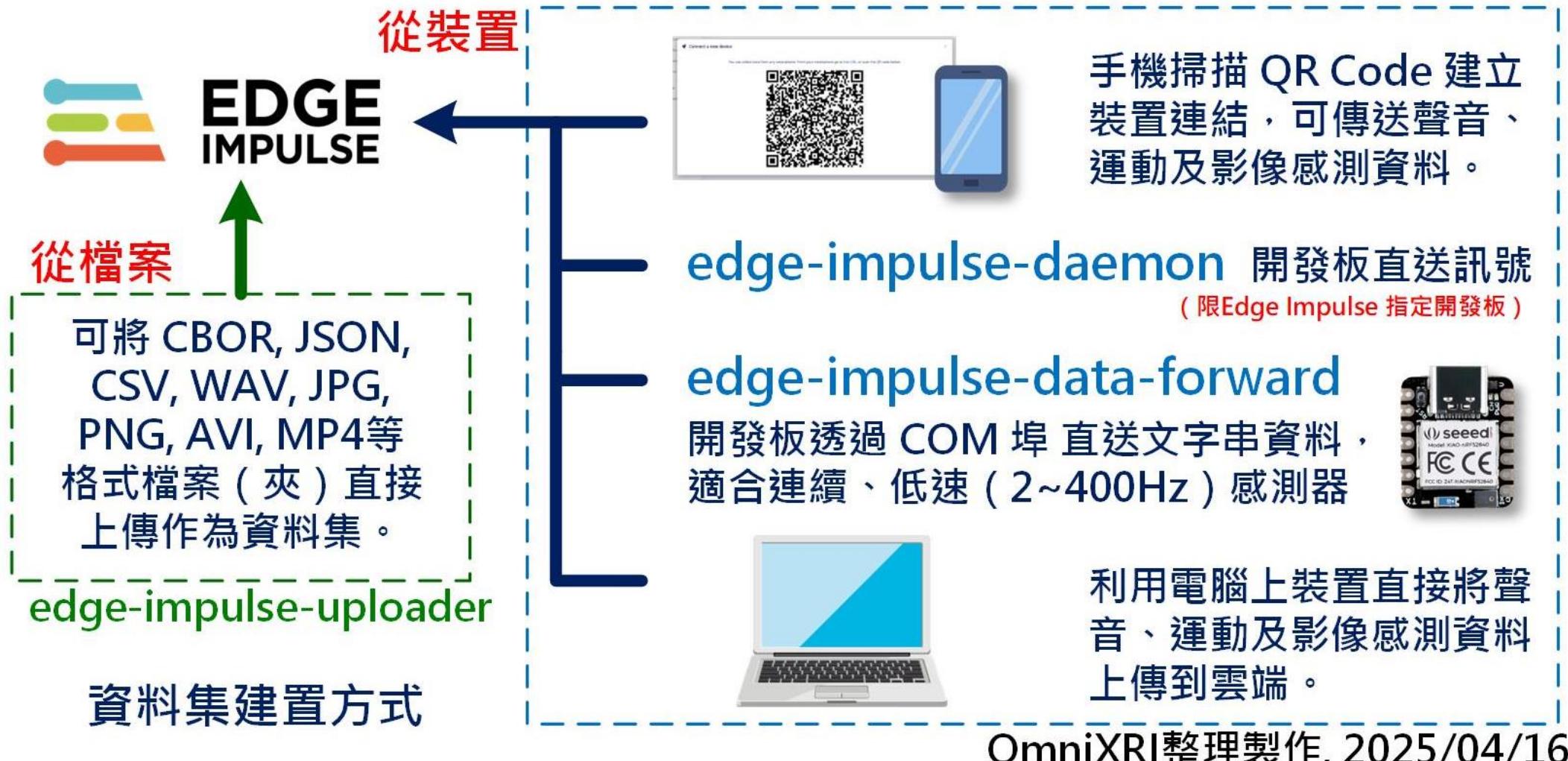
Medium Edge
Mini Edge
Micro Edge

沒有最邊
只有更邊

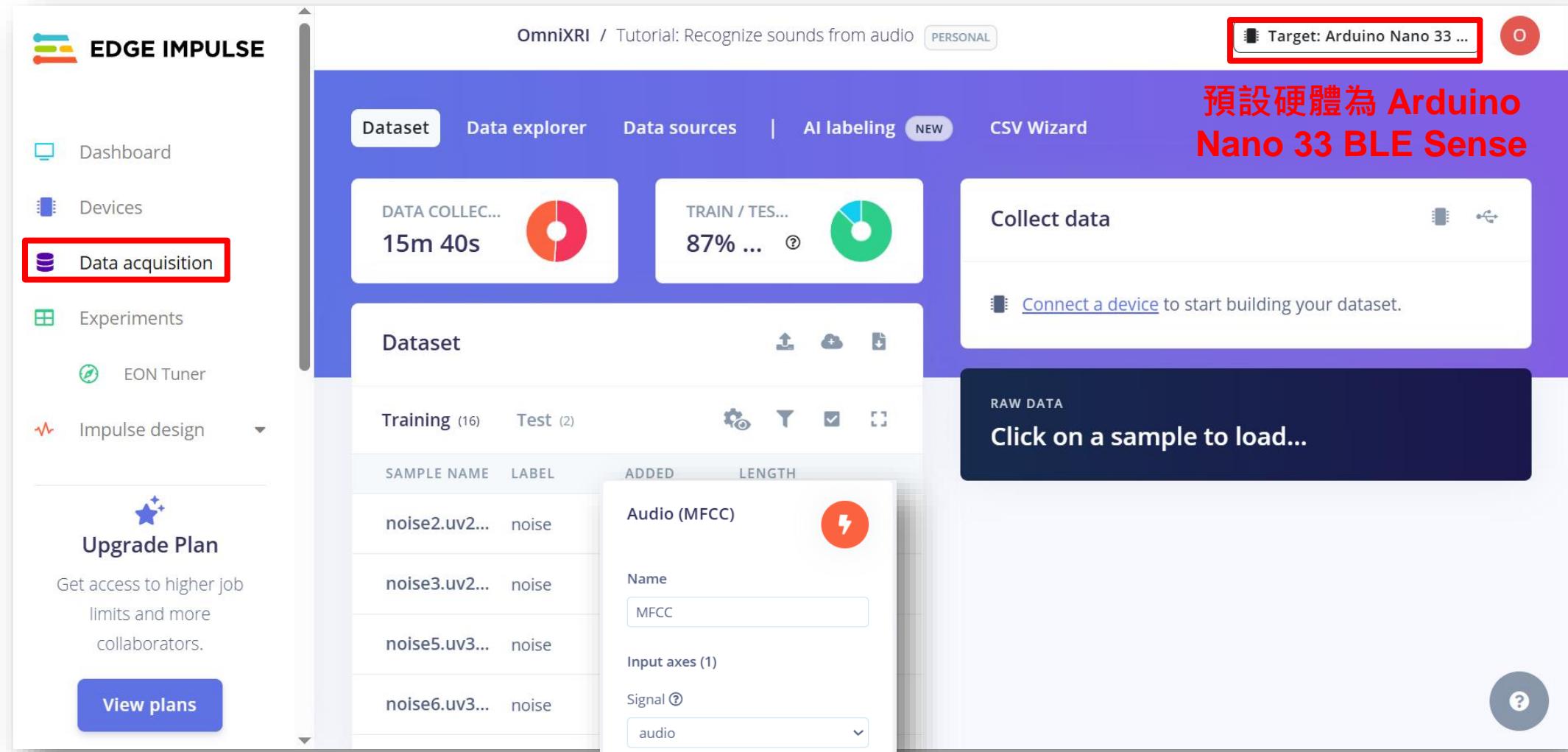
- 3.1. 資料集建置
- 3.2. 公開資料集
- 3.3. 資料集標註
- 3.4. 資料集迷思

OmniXRI's Edge AI & TinyML 小學堂 (2024) 【第3講】 資料集建置與標註
<https://youtu.be/d655nS-0XmM>

Edge Impulse 資料集建置方式



水流聲辨識範例 — 資料收集



The screenshot shows the Edge Impulse web interface for data acquisition. On the left sidebar, the 'Data acquisition' option is highlighted with a red box. The main dashboard displays a dataset summary: 'DATA COLLEC...' (15m 40s) and 'TRAIN / TES...' (87%). A large callout box on the right side of the screen, also with a red border, states '預設硬體為 Arduino Nano 33 BLE Sense'. Below this, a 'Collect data' section prompts the user to 'Connect a device to start building your dataset.' At the bottom, a dark blue box says 'RAW DATA Click on a sample to load...'. A modal window is open over the dataset table, showing details for an 'Audio (MFCC)' signal: 'Name' is set to 'MFCC', 'Input axes (1)' is selected, and 'Signal' is set to 'audio'. The dataset table lists four samples: 'noise2.uv2...' (noise), 'noise3.uv2...' (noise), 'noise5.uv3...' (noise), and 'noise6.uv3...' (noise).



6.2. 模型選用與訓練

選擇模型及設定必要參數

OmniXRI / Tutorial: Recognize sounds from audio PERSONAL Target: Arduino Nano 33 ... O

Impulse #1

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

Time series data

- Input axes: audio
- Window size: 1,000 ms.
- Window increase (stride): 300 ms.
- Frequency (Hz): 16000
- Zero-pad data: checked

Spectrogram

- Name: Spectrogram
- Input axes (1): audio

Classification (Keras)

- Name: NN Classifier
- Input features: Spectrogram, MFCC
- Output features: 2 (faucet, noise)

Output features

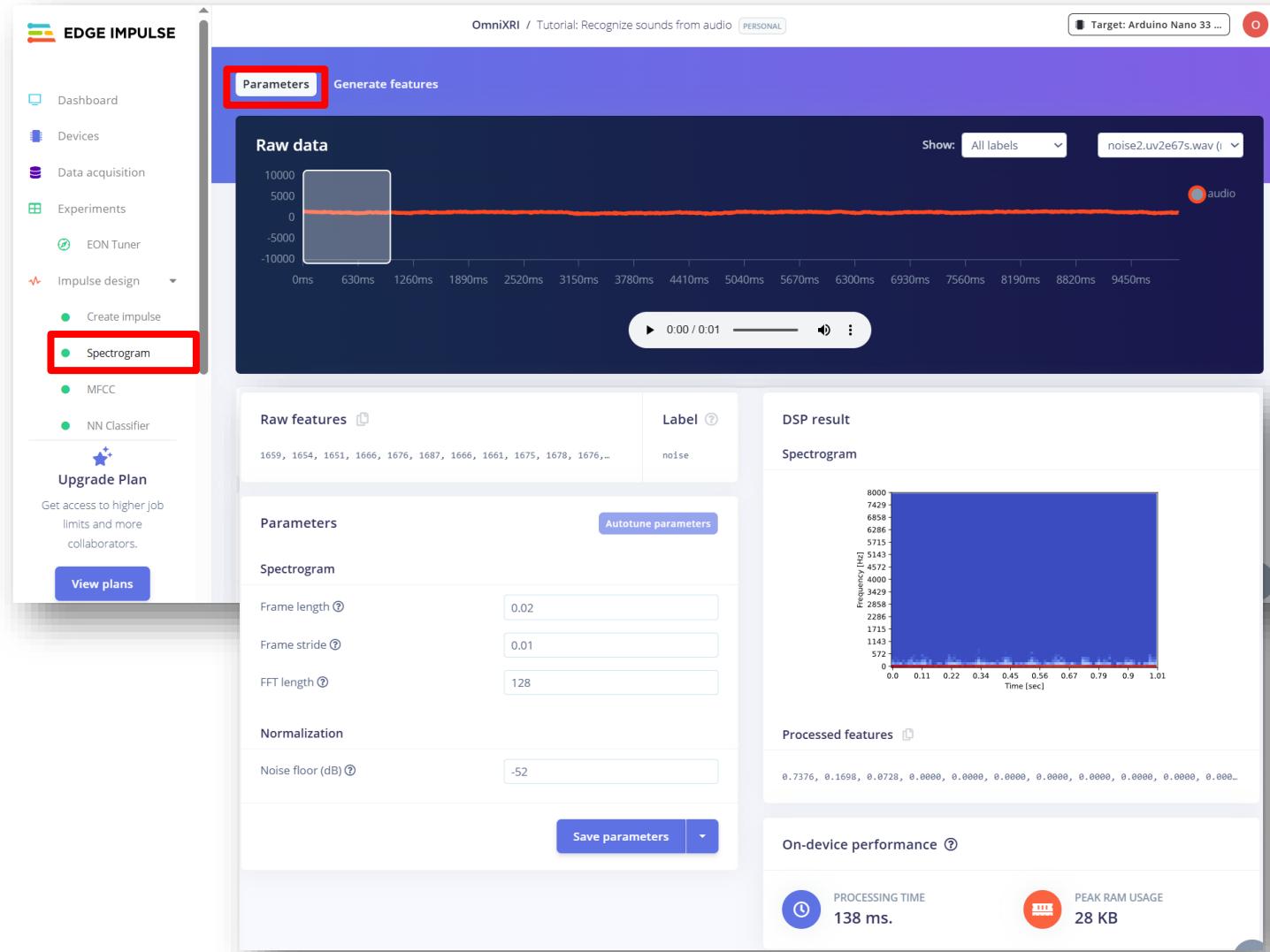
- 2 (faucet, noise)

Save Impulse

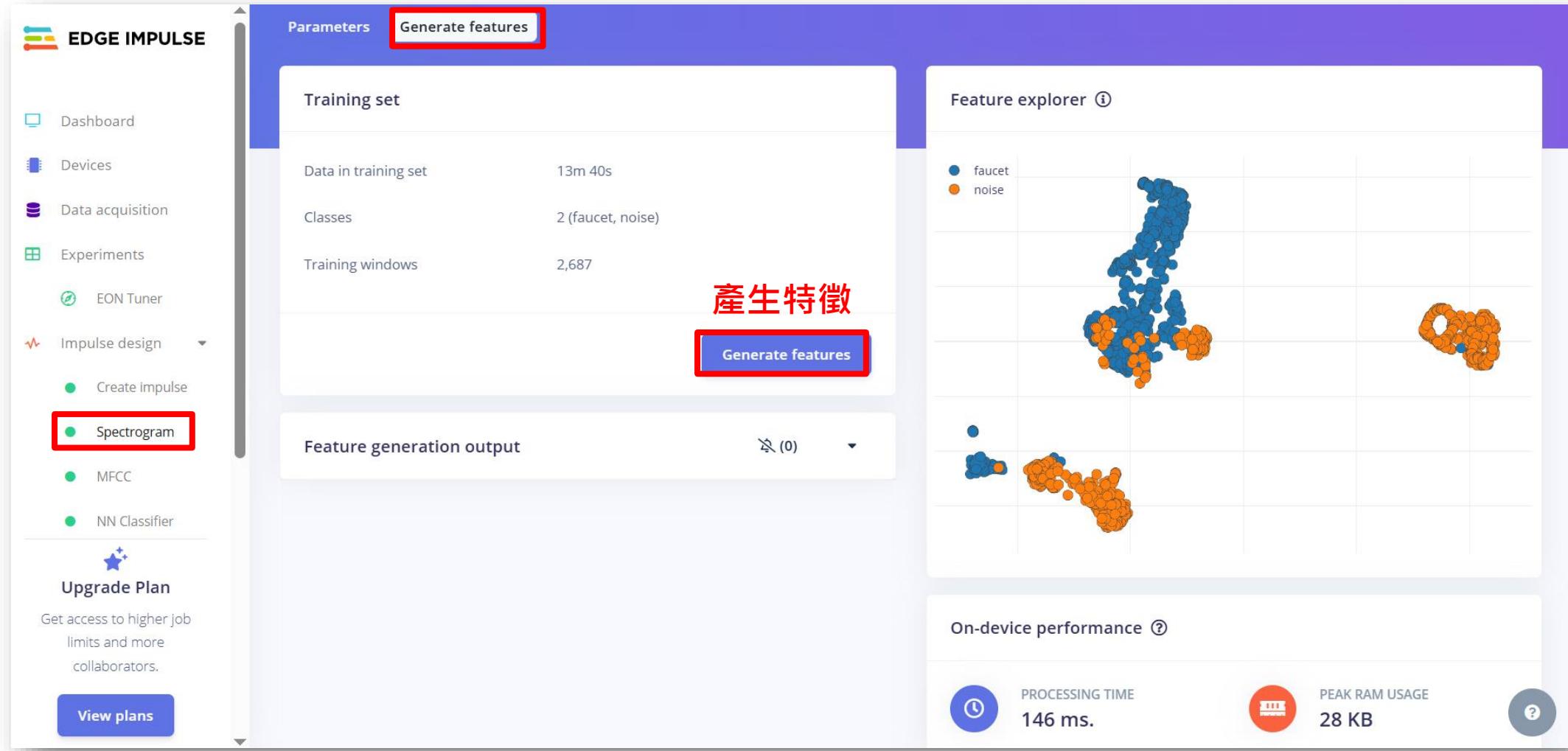
修改後記得存檔

可依需求自行新增修改 Block

提取資料特徵 - Spectrogram



產生特徵結果 - Spectrogram

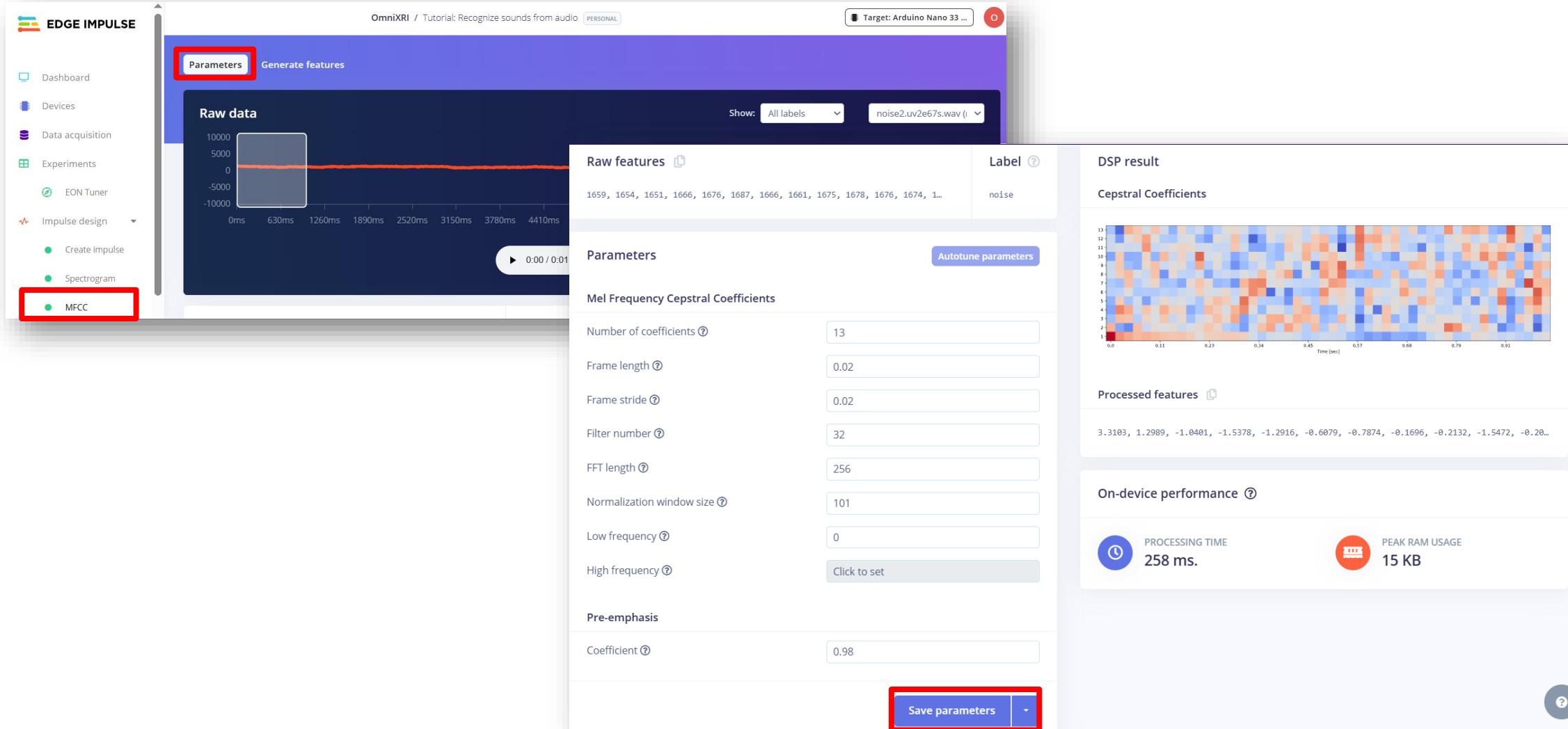


The screenshot shows the Edge Impulse web interface for generating features from a spectrogram. The left sidebar includes options like Dashboard, Devices, Data acquisition, Experiments, EON Tuner, Impulse design, Create impulse, Spectrogram (which is highlighted with a red box), MFCC, and NN Classifier. The main area has tabs for Parameters and Generate features (also highlighted with a red box). Under Parameters, the Training set section displays:

- Data in training set: 13m 40s
- Classes: 2 (faucet, noise)
- Training windows: 2,687

A large red button labeled "產生特徵" (Generate features) is centered below the training set section. The Feature explorer section contains a scatter plot with blue dots representing "faucet" and orange dots representing "noise". The On-device performance section at the bottom right shows a processing time of 146 ms and peak RAM usage of 28 KB.

提取資料特徵 - MFCC



The screenshot shows the Edge Impulse Parameters page for generating features from raw audio data. The left sidebar highlights the 'MFCC' option under the 'Create impulse' section. The main interface displays a spectrogram titled 'Raw data' and a list of 'Raw features' labeled 'noise'. The 'Parameters' section is detailed below:

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
- Low frequency: 0
- High frequency: Click to set

Pre-emphasis

- Coefficient: 0.98

Save parameters

The right side of the screen shows the 'DSP result' section, which includes a heatmap titled 'Cepstral Coefficients' showing spectral patterns over time. Below it are sections for 'Processed features' (listing numerical values) and 'On-device performance' (showing processing time as 258 ms and peak RAM usage as 15 KB).

產生特徵結果 - MFCC

OmniXRI / Tutorial: Recognize sounds from audio PERSONAL Target: Arduino Nano 33 ... 0

EDGE IMPULSE

- Dashboard
- Devices
- Data acquisition
- Experiments
- EON Tuner
- Impulse design
 - Create impulse
 - Spectrogram
 - MFCC**
 - NN Classifier
- Retrain model

Upgrade Plan
Get access to higher job limits and more collaborators.
[View plans](#)

Parameters **Generate features** (highlighted)

Training set

Data in training set	13m 40s
Classes	2 (faucet, noise)
Training windows	2,687

Feature explorer (highlighted)

faucet (blue dots), noise (orange dots)

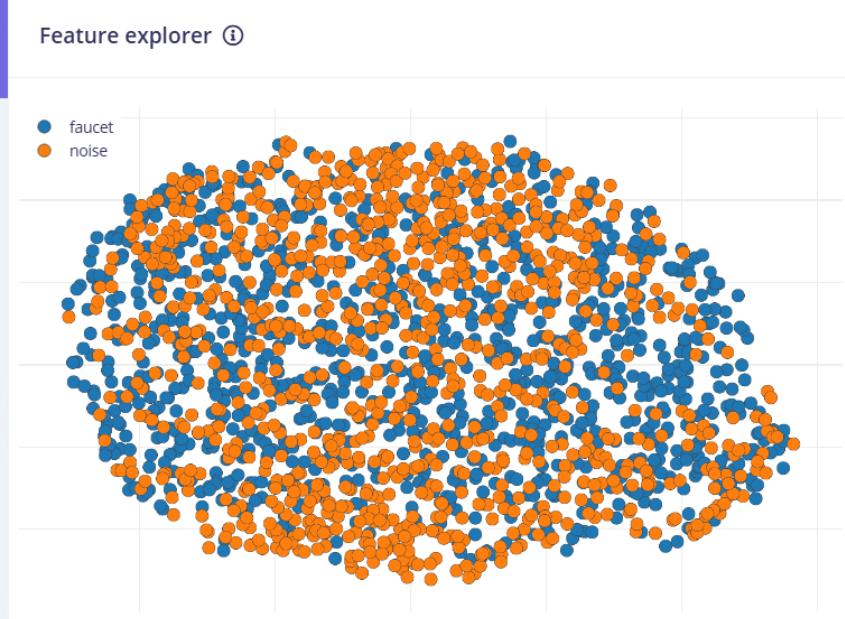
產生特徵 (highlighted)

Feature generation output

Click 'Generate features' to start
View output from previous jobs: <https://studio.edgeimpulse.com/studio/391938/jobs?key=dsp-studio-wrupper-29>

On-device performance

PROCESSING TIME: 266 ms. PEAK RAM USAGE: 15 KB



設定分類訓練相關參數

EDGE IMPULSE

- Dashboard
- Devices
- Data acquisition
- Experiments
- EON Tuner
- Impulse design
 - Create impulse
 - Spectrogram
 - MFCC
 - NN Classifier

神經網路參數設定

Neural Network settings

Training settings

Number of training cycles ② **訓練次數**

Use learned optimizer ②

Learning rate ② **學習率**

Training processor ② **訓練處理器**

Advanced training settings

Validation set size ② **驗證集比例** %

Split train/validation set on metadata key ②

Batch size ② **批次大小**

Auto-weight classes ②

Profile int8 model ②

神經網路結構設定

Neural network architecture

```

graph TD
    Input[Input layer (7,085 features)] --> Reshape[Reshape layer (65 columns)]
    Reshape --> Conv1[1D conv / pool layer (8 filters, 3 kernel size, 1 layer)]
    Conv1 --> Dropout1[Dropout (rate 0.25)]
    Dropout1 --> Conv2[1D conv / pool layer (16 filters, 3 kernel size, 1 layer)]
    Conv2 --> Dropout2[Dropout (rate 0.25)]
    Dropout2 --> Flatten[Flatten layer]
    Flatten -.-> AddLayer[Add an extra layer]
    AddLayer -.-> Output[Output layer (2 classes)]
  
```

開始訓練

模型訓練結果

Training output

Model Model version: ② Quantized (int8) ▾

Last training performance (validation set)

 ACCURACY	99.4%	 LOSS	0.02
--	-------	--	------

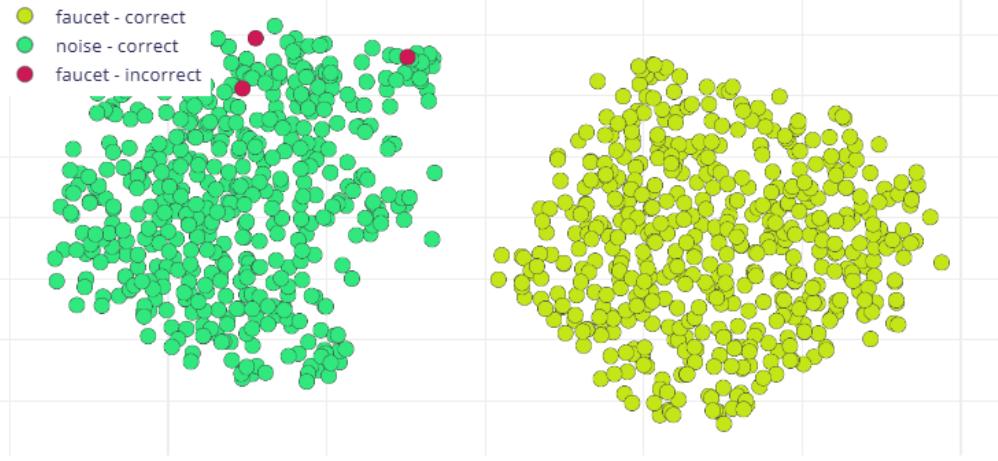
Confusion matrix (validation set)

	FAUCET	NOISE
FAUCET	99.3%	0.7%
NOISE	0.4%	99.6%
F1 SCORE	0.99	0.99

Metrics (validation set)

METRIC	VALUE
Area under ROC Curve ②	0.99
Weighted average Precision ②	0.99
Weighted average Recall ②	0.99
Weighted average F1 score ②	0.99

Data explorer (full training set) ②



On-device performance ②

Engine: ② EON™ Compiler (RAM optimized) ▾

 INFERENCING TIME	16 ms.	 PEAK RAM USAGE	22.0K	 FLASH USAGE	42.9K
--	--------	--	-------	---	-------

線上測試（從內建測試集）

- Impulse design
- Create impulse
- Spectrogram
- MFCC
- NN Classifier
- ✖ Retrain model
- **Live classification**
- ✓ Model testing
- Perf. calibration
- Deployment

Classify existing test sample

Load sample

Classification result

Summary		Model version: Unoptimized (float32)	⋮
Name	<input type="text" value="faucet1.v1m71av.wav"/>		
Label	<input type="text" value="faucet"/>		
CATEGORY	COUNT		
faucet	197		
noise	0		
uncertain	0		

Detailed result

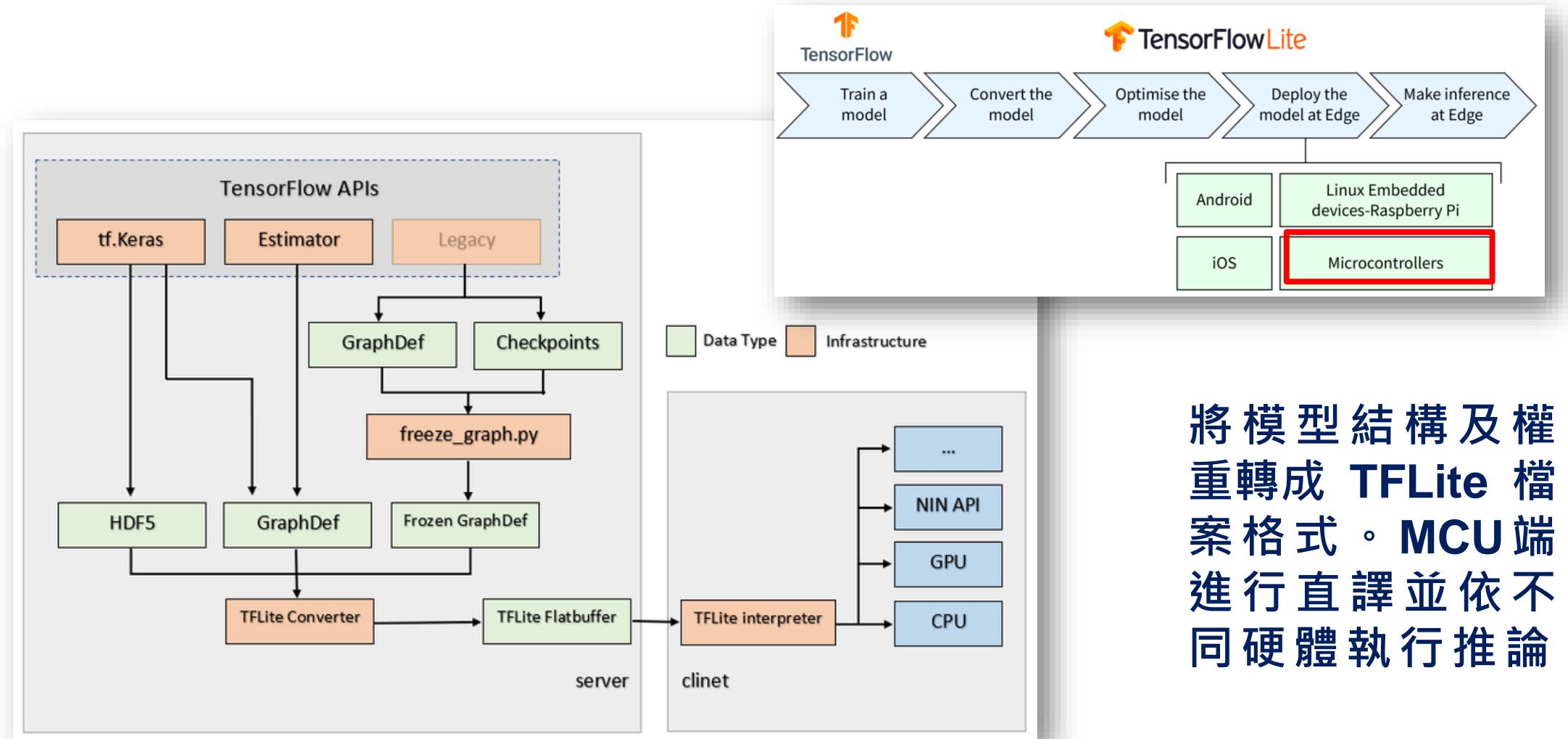
Show only unknowns

TIMESTAMP	FAUCET	NOISE
0	1.00	0
300	1.00	0
600	1.00	0
900	1.00	0
1,200	1.00	0
1,500	1.00	0
1,800	1.00	0
2,100	1.00	0
2,400	1.00	0
2,700	1.00	0
3,000	1.00	0



6.3. 模型優化及部署

TensorFlow Lite 轉換程序



目標優化 – EON Tuner (AutoML)

The EON Tuner helps you find the most optimal architecture for your embedded machine-learning application.

New run

➤ 時間優先
➤ 儲存優先
➤ 記憶體優先
➤ 精度優先

Target

- No name set
- Arduino Nano 33 BLE Sense (Cortex-M4F 64MHz)
- 100 ms
- 256 kB
- 1024 kB

Filters

Status	Count
Pending	0
Running	0
Completed	50
Failed	0

DSP type	Count
MFCC	7
MFE	16
Spectrogram	27

Model type	Count
Convolutional (1D)	40
Convolutional (2D)	4

Experiments **EON Tuner**

EDGE IMPULSE

- Dashboard
- Devices
- Data acquisition
- Experiments
- EON Tuner**
- Impulse design
 - Create impulse
 - Spectrogram
 - MFCC
 - NN Classifier
- Retrain model

Upgrade Plan

Get access to higher job limits and more collaborators.

View plans

Performance Metrics

- spectr-conv1d-9b0**: Latency 15 ms of 100 ms, RAM 17 kB of 256 kB, ROM 41 kB of 1024 kB
- mfe-conv1d-e4f**: Latency 21 ms of 100 ms, RAM 20 kB of 256 kB
- spectr-conv1d-be7**: Latency 84 ms of 100 ms, RAM 21 kB of 256 kB, ROM 47 kB of 1024 kB

Accuracy (Classification)

Model	Class	Accuracy (%)
spectr-conv1d-9b0	fau	100
	noi	0
mfe-conv1d-e4f	fau	0
	noi	100
spectr-conv1d-be7	fau	1
	noi	1

TIME-SERIES INPUT

- 1000 ms | 1000 ms | Enabled

SPECTROGRAM

- 0.05 | 0.05 | -72

ACCURACY (CLASSIFICATION)

模型優化

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

 TensorFlow Lite

Quantized (int8) 整數8位元

	SPECTROGRAM	MFCC	NN CLASSIFIER	TOTAL
LATENCY	146 ms.	266 ms.	16 ms.	428 ms.
RAM	27.8K	15.4K	21.5K	27.8K
FLASH	-	-	53.5K	-
ACCURACY				-

Unoptimized (float32) 浮點32位元

	SPECTROGRAM	MFCC	NN CLASSIFIER	TOTAL
LATENCY	146 ms.	266 ms.	467 ms.	879 ms.
RAM	27.8K	15.4K	71.0K	71.0K
FLASH	-	-	60.7K	-
ACCURACY				-

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

Estimate for Arduino Nano 33 BLE Sense (Cortex-M4F 64MHz) - [Change target](#)

開始建構 **Build**

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

 EON™ Compiler
Same accuracy, 21% less RAM, 44% less ROM.

Quantized (int8)

	SPECTROGRAM	MFCC	NN CLASSIFIER	TOTAL
LATENCY	146 ms.	266 ms.	16 ms.	428 ms.
RAM	27.8K	15.4K	17.0K	27.8K
FLASH	-	-	33.1K	-
ACCURACY				-

Unoptimized (float32)

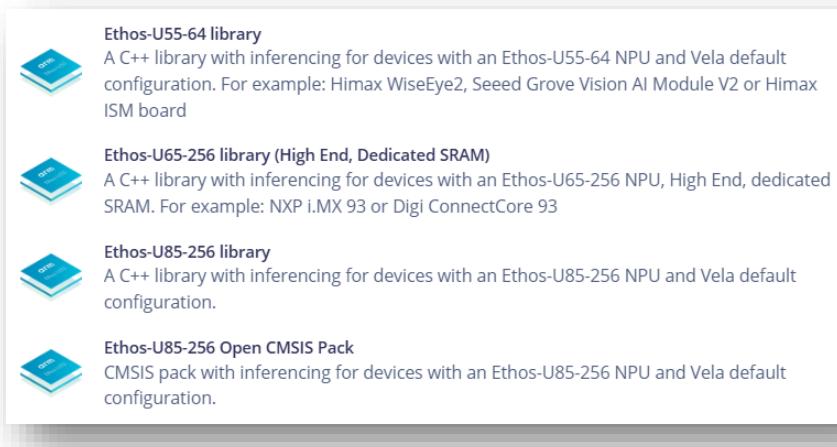
	SPECTROGRAM	MFCC	NN CLASSIFIER	TOTAL
LATENCY	146 ms.	266 ms.	467 ms.	879 ms.
RAM	27.8K	15.4K	57.3K	57.3K
FLASH	-	-	33.8K	-
ACCURACY				-

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

Estimate for Arduino Nano 33 BLE Sense (Cortex-M4F 64MHz) - [Change target](#)

開始建構 **Build**

可部署方式



1. 客製化函式庫

C++, **Arduino**, WebAssembly, Cube.MX CMSIS-Pack, DAR-AI, OpenMV, **Ethos-U (NPU)**, Simplicity Studio, TensorRT, TIDL, Tensai Flow

2. 預編譯韌體

搭配 edge-impulse-run-impulse 使用

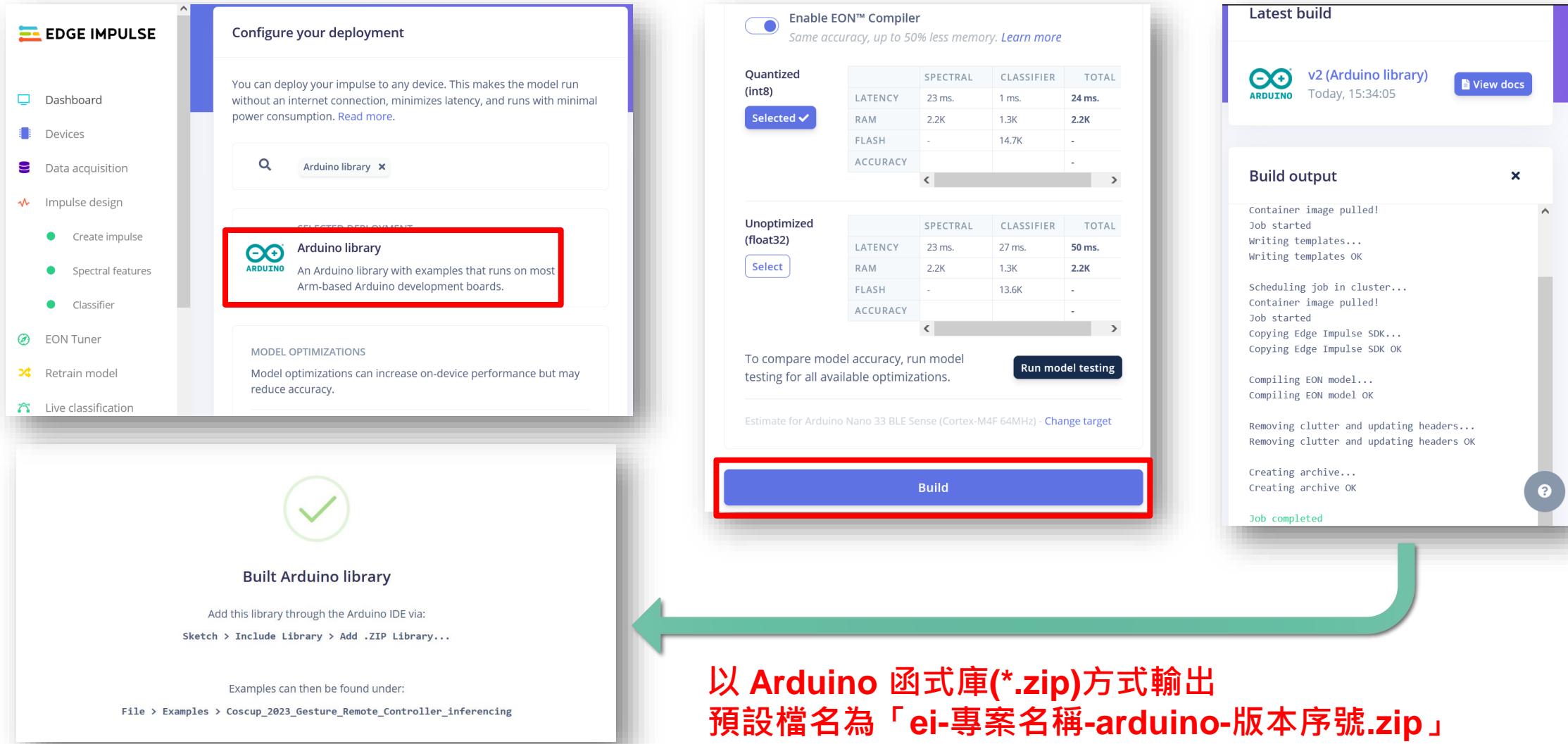
3. Linux 型式

4. Docker 型式

5. 在手機或電腦執行

6. 客製化部署方塊（企業版）

以 Arduino 型式輸出為例



The screenshot shows the Edge Impulse deployment configuration interface. On the left, under 'Deployment', the 'Arduino library' is selected (highlighted with a red box). In the center, the 'Build' button is highlighted with a red box. On the right, the 'Latest build' panel shows the build process for the 'v2 (Arduino library)' version, which was completed today at 15:34:05.

Configure your deployment

You can deploy your impulse to any device. This makes the model run without an internet connection, minimizes latency, and runs with minimal power consumption. [Read more](#).

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.

Enable EON™ Compiler
Same accuracy, up to 50% less memory. [Learn more](#)

Quantized (int8)
Selected ✓

	SPECTRAL	CLASSIFIER	TOTAL
LATENCY	23 ms.	1 ms.	24 ms.
RAM	2.2K	1.3K	2.2K
FLASH	-	14.7K	-
ACCURACY	-		

Unoptimized (float32)
Select

	SPECTRAL	CLASSIFIER	TOTAL
LATENCY	23 ms.	27 ms.	50 ms.
RAM	2.2K	1.3K	2.2K
FLASH	-	13.6K	-
ACCURACY	-		

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

Estimate for Arduino Nano 33 BLE Sense (Cortex-M4F 64MHz) - [Change target](#)

Build

Latest build

v2 (Arduino library) Today, 15:34:05 [View docs](#)

Build output

- Container image pulled!
- Job started
- Writing templates...
- Writing templates OK
- Scheduling job in cluster...
- Container image pulled!
- Job started
- Copying Edge Impulse SDK...
- Copying Edge Impulse SDK OK
- Compiling EON model...
- Compiling EON model OK
- Removing clutter and updating headers...
- Removing clutter and updating headers OK
- Creating archive...
- Creating archive OK
- Job completed**

Built Arduino library

Add this library through the Arduino IDE via:
Sketch > Include Library > Add .ZIP Library...

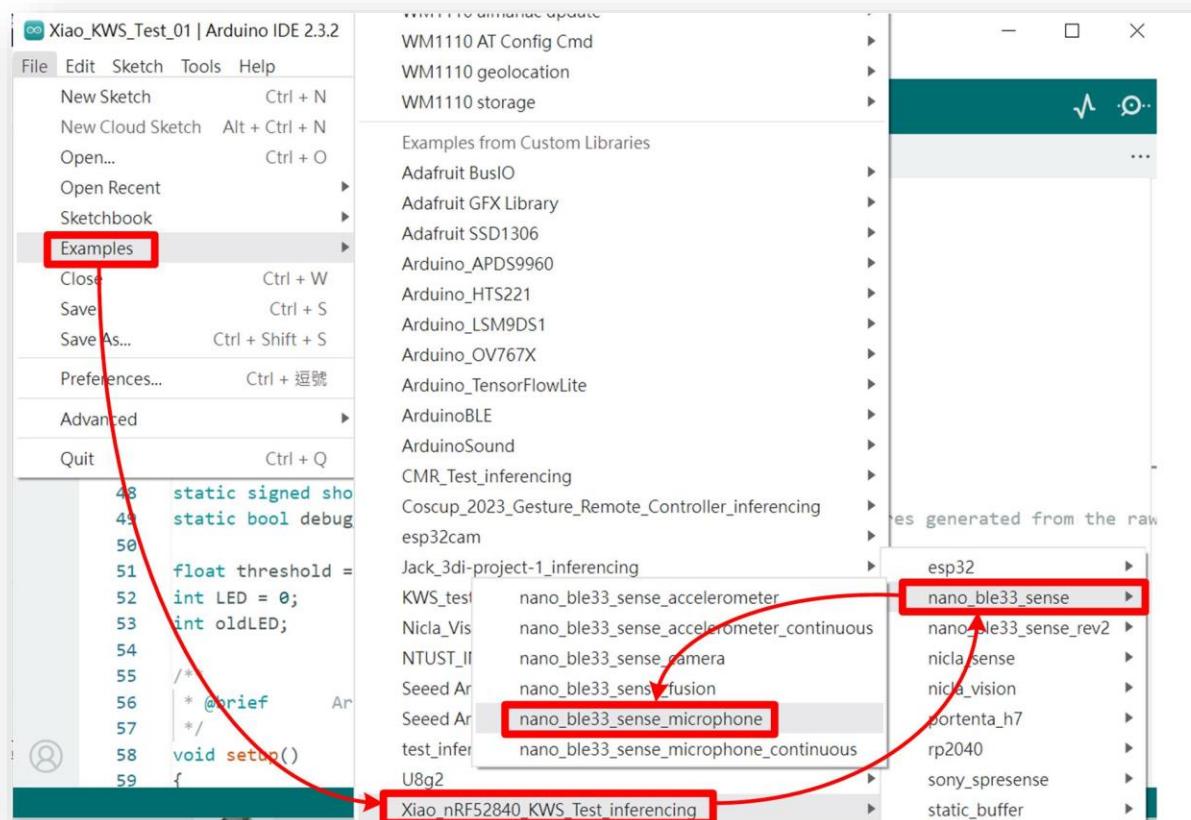
Examples can then be found under:
File > Examples > Coscup_2023_Gesture_Remote_Controller_inferencing

**以 Arduino 函式庫(*.zip)方式輸出
預設檔名為「ei-專案名稱-arduino-版本序號.zip」**

導入 Arduino 函式庫並進行推論測試

新增函式庫 Sketch > Include Library > Add .ZIP Library ...

新增範例 File > Examples > ei-專案名稱_inferencing > nano_ble33_sense > nano_ble33_sense_microphone (原始範例輸出結果從 COM 輸出文字串)



```

51 float threshold = 0.50;
52 int LED = 0;
53 int oldLED;

117 //lets light up some LEDs
118 if (result.classification[ix].value > threshold) {
119 //now let's see what label were in
120 switch (ix) {
121 case 0: LED = 11; break; // RED LED
122 case 1: LED = 12; break; // BLUE LED
123 case 2: LED = 13; break; // GREEN LED
124 default: LED = 0;
125 }
126 //in Sense, LOW will light up the LED
127 if (LED != 0) {
128   digitalWrite (oldLED, HIGH); //if we enter a wo
129   digitalWrite (LED, LOW);
130   oldLED = LED;
131 }
132 else //turn off LED
133   digitalWrite (oldLED, HIGH);
134 }
```

新增程式碼 (Low Code)
可調整觸發門檻
可指定辨完成亮滅LED

參考文獻

- 許哲豪，臺灣科技大學資訊工程系「人工智慧與邊緣運算實務」(2021~2023)
<https://omnixri.blogspot.com/p/ntust-edge-ai.html>
- 許哲豪，OmniXRI's Edge AI & TinyML 小學堂 Youtube 直播課程總結
<https://omnixri.blogspot.com/2024/06/omnixris-edge-ai-tinyml-youtube.html>
- 許哲豪，歐尼克斯實境互動工作室系列發文—TinyML(MCU AI)系列
<https://hackmd.io/1PK1URhIQ7GutcWgpgsWbg#TinyMLMCU-AI%E7%B3%BB%E5%88%97>
- Edge Impulse Studio Document
<https://docs.edgeimpulse.com/docs>
- 許哲豪，【課程簡報】20240509_慈濟醫資_穿戴式人工智慧工作坊_利用TinyML技術快速搭建微型智慧應用
<https://omnixri.blogspot.com/2024/05/20240509tinyml12.html>

沒有最邊



只有更邊



歡迎加入
邊緣人俱樂部



YOUTUBE 直播 : <https://www.youtube.com/@omnixri1784streams>



歐尼克斯實境互動工作室
(OmniXRI Studio)

許哲豪 (Jack Hsu)

[Facebook : Jack Omnixri](#)

[FB社團 : Edge AI Taiwan 邊緣智能交流區](#)

[電子郵件 : omnixri@gmail.com](#)

[部落格 : https://omnixri.blogspot.tw](#)

[開 源 : https://github.com/OmniXRI](#)