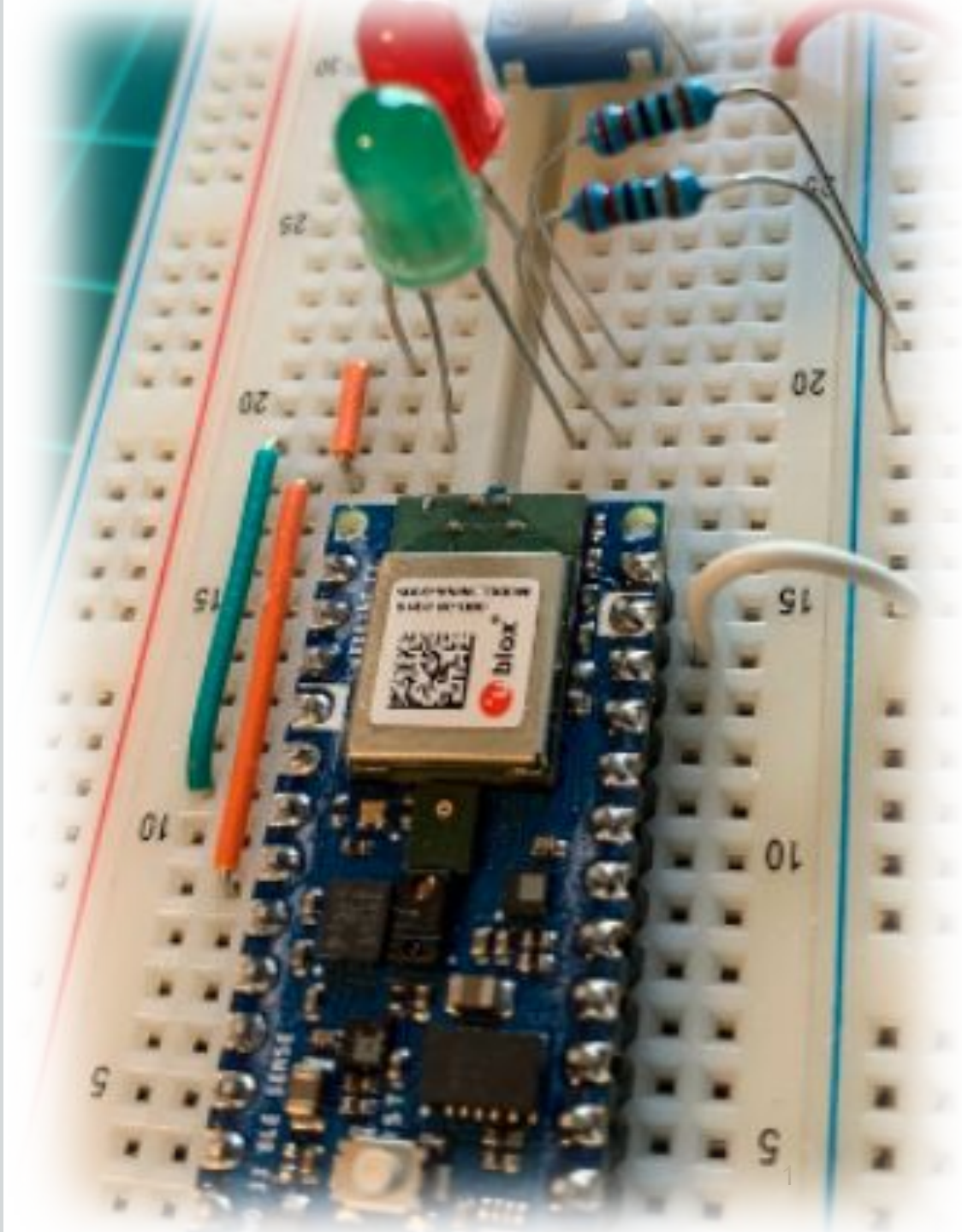
IESTI01 – TinyML

Embedded Machine Learning

29. EdgeAI Going Further



Prof. Marcelo Rovai
UNIFEI

EdgeAI

Going Further

Definitions

- **Artificial Intelligence (AI)**: The broad discipline of creating **machines capable of performing tasks that typically require human intelligence**, including learning, reasoning, and problem-solving.
- **Machine Learning (ML)**: A subset of AI focused on building **systems that learn from data**, allowing computers to improve their performance on tasks without being explicitly programmed for each task.
- **Deep Learning (DL)**: An advanced **subset of machine learning involving neural networks** with multiple layers (deep networks), enabling powerful data modeling and decision-making based on vast amounts of data.
- **Edge AI**: The practice of **processing AI algorithms on local devices near the data source**, reducing latency and bandwidth use while enhancing privacy and speed in applications like real-time analytics and local data processing.
- **TinyML**: The field of **machine learning focused on developing low-power models** that can operate directly on small devices like microcontrollers, extending AI capabilities to edge devices.

Hardware

Power

EdgeML (AI)

TinyML

Video Classification 2 MB+

Object Detection Complex Voice Processing 1 MB+

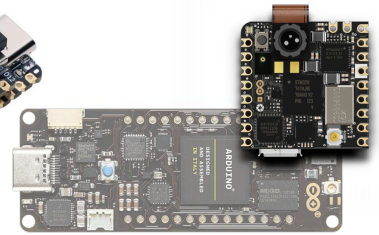
Image Classification 250 KB+

KeyWord Spotting Audio Classification 50 KB

Anomaly Detection Sensor Classification 20 KB

Wio

ESP32S3



Rpi-Pico (Cortex-M0+)

Arduino Nano (Cortex-M4)

Arduino Pro (Cortex-M7)

RaspberryPi SmartPhone (Cortex-A)

Jetson Nano (Cortex-A + GPU)

Source: Edge Impulse

Software

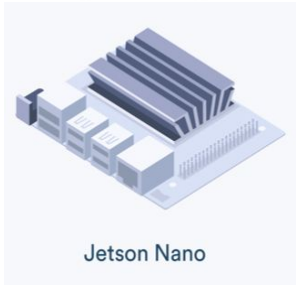
TensorFlow

TensorFlow Lite

Train a model → Convert model → Optimize model → Deploy model at Edge → Make inferences at Edge

EdgeAI

Raspberry Pi



Linux

iOS

TinyML

TensorFlow Lite Micro

Microcontroller

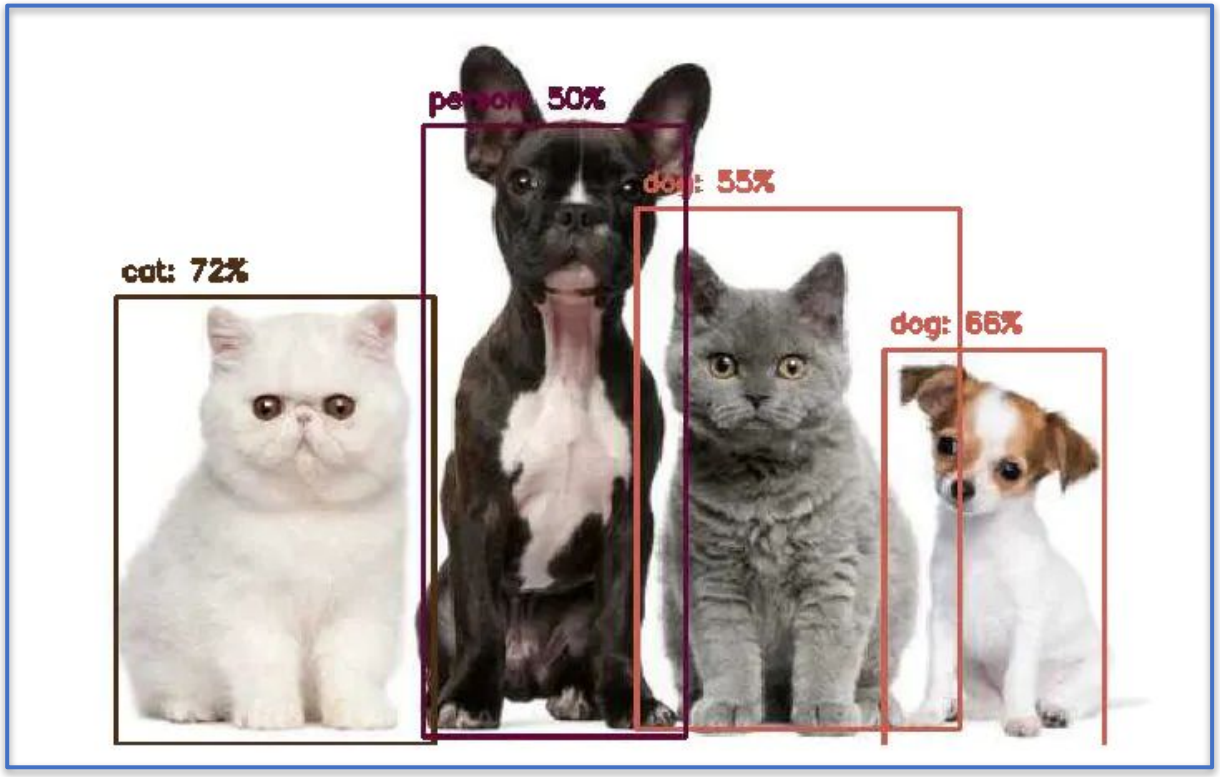
Computer Vision Main Types

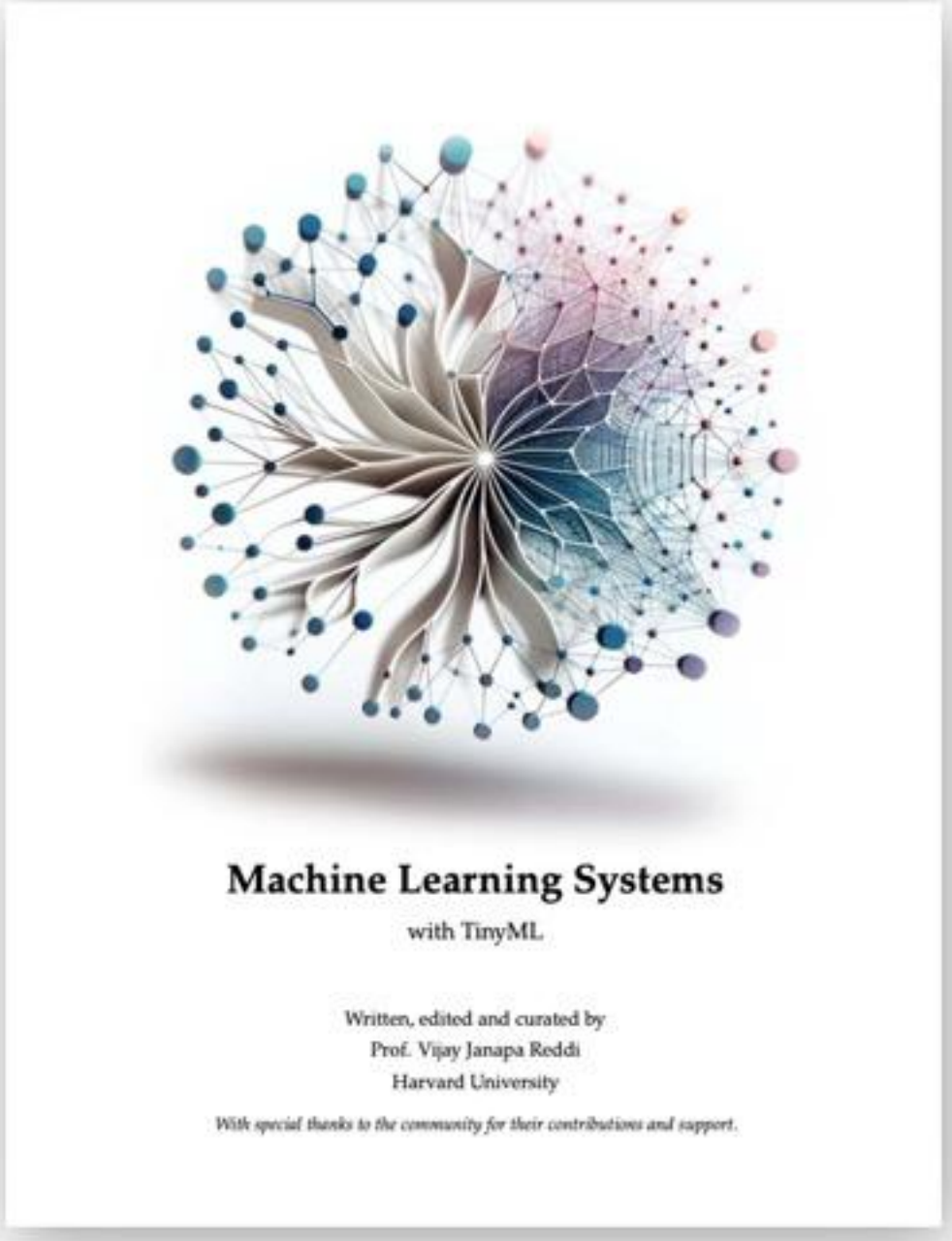
Image Classification
(**Multi-Class** Classification)

Cat: 70%

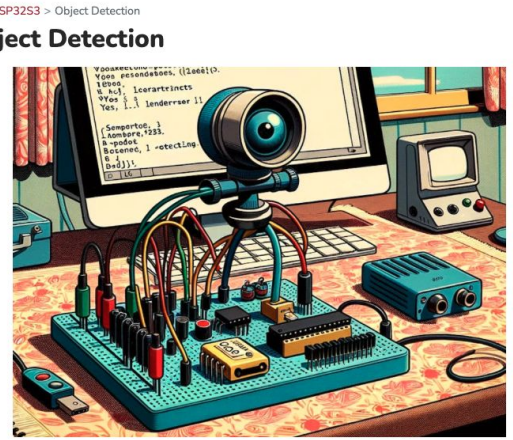
Dog: 80%

Object Detection
Multi-Label Classification + Object **Localization**





Nicla Vision



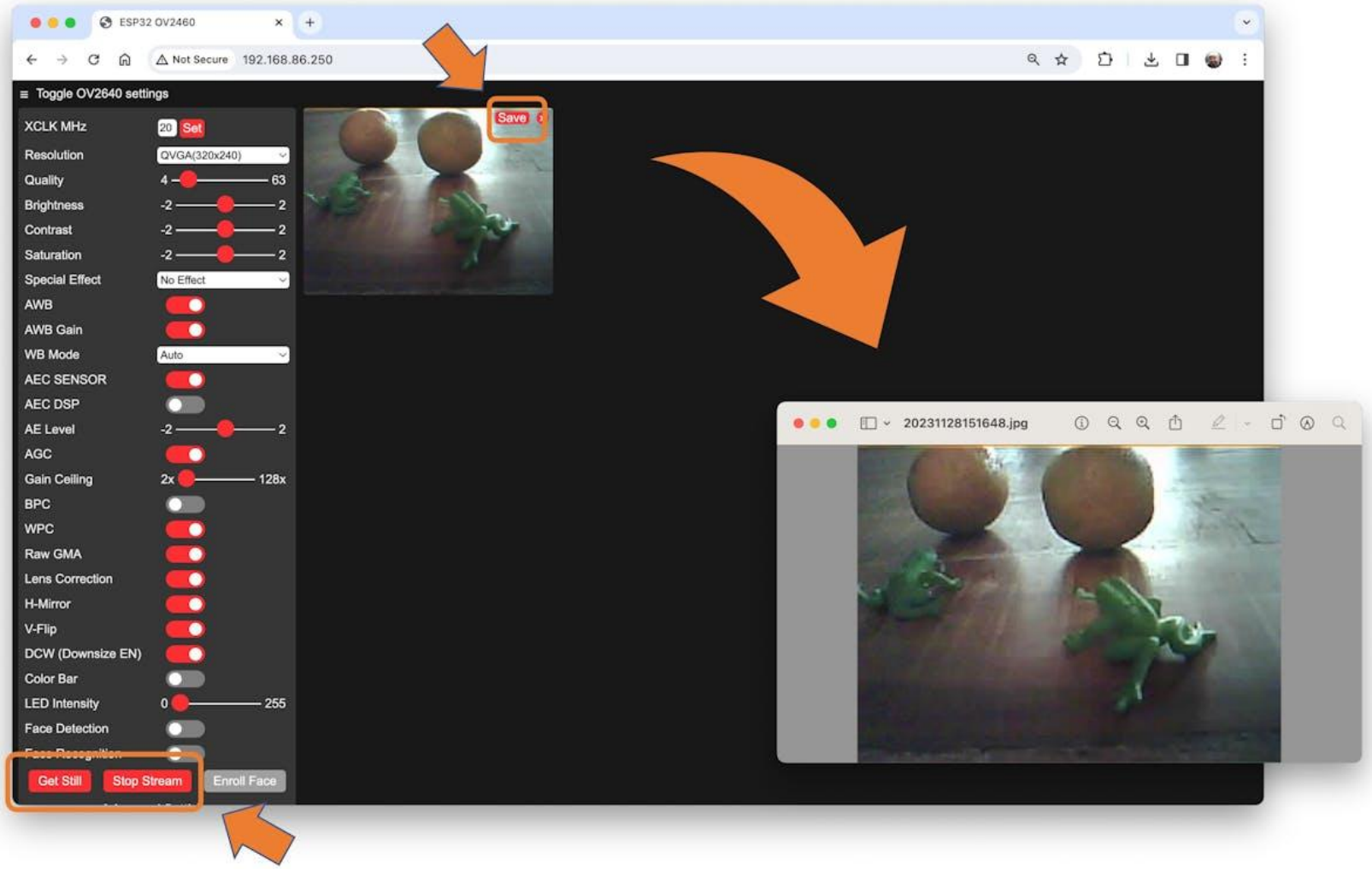
XIAO ESP32S3

FOMO

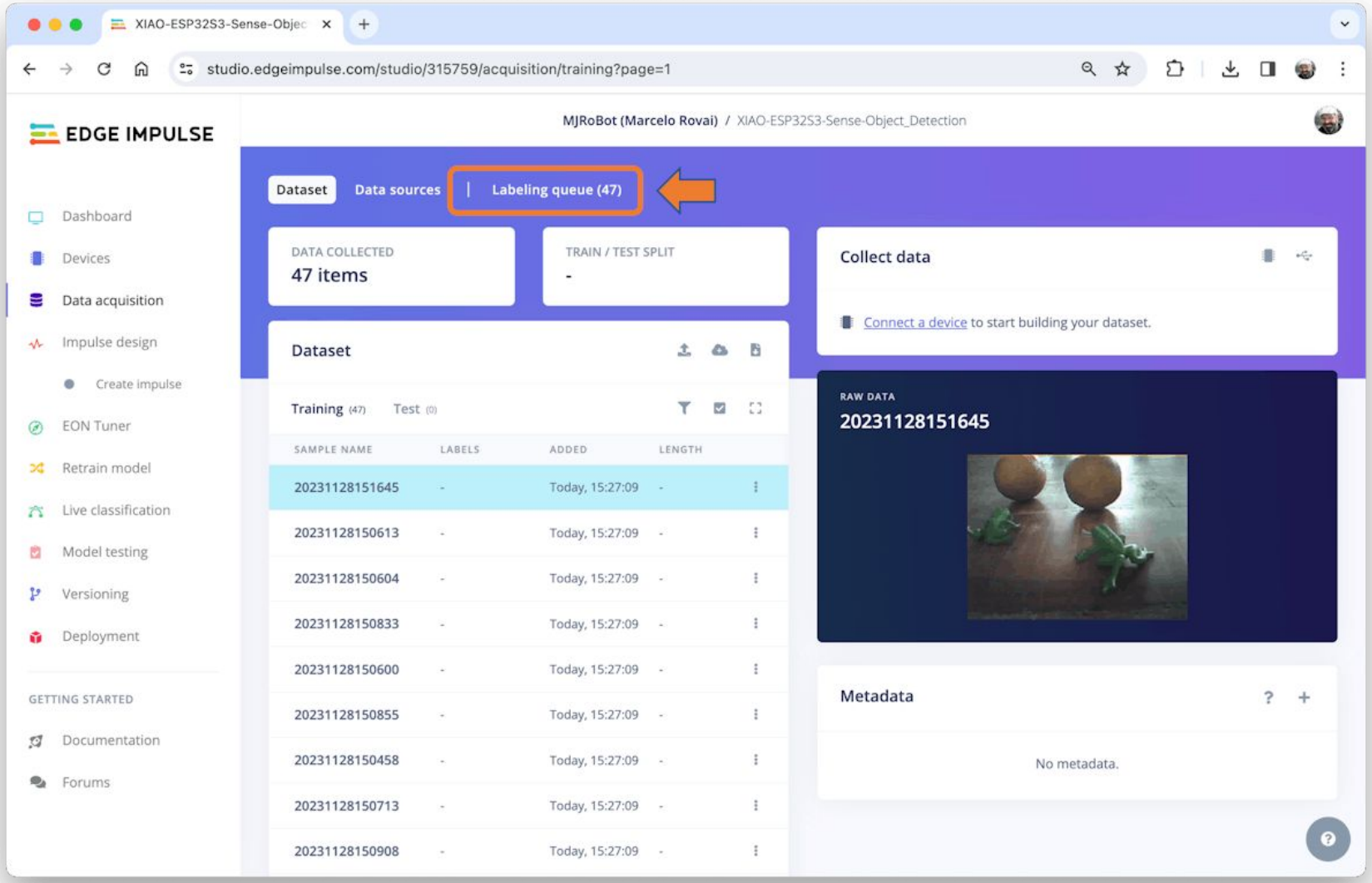
Object Detection model

Dataset

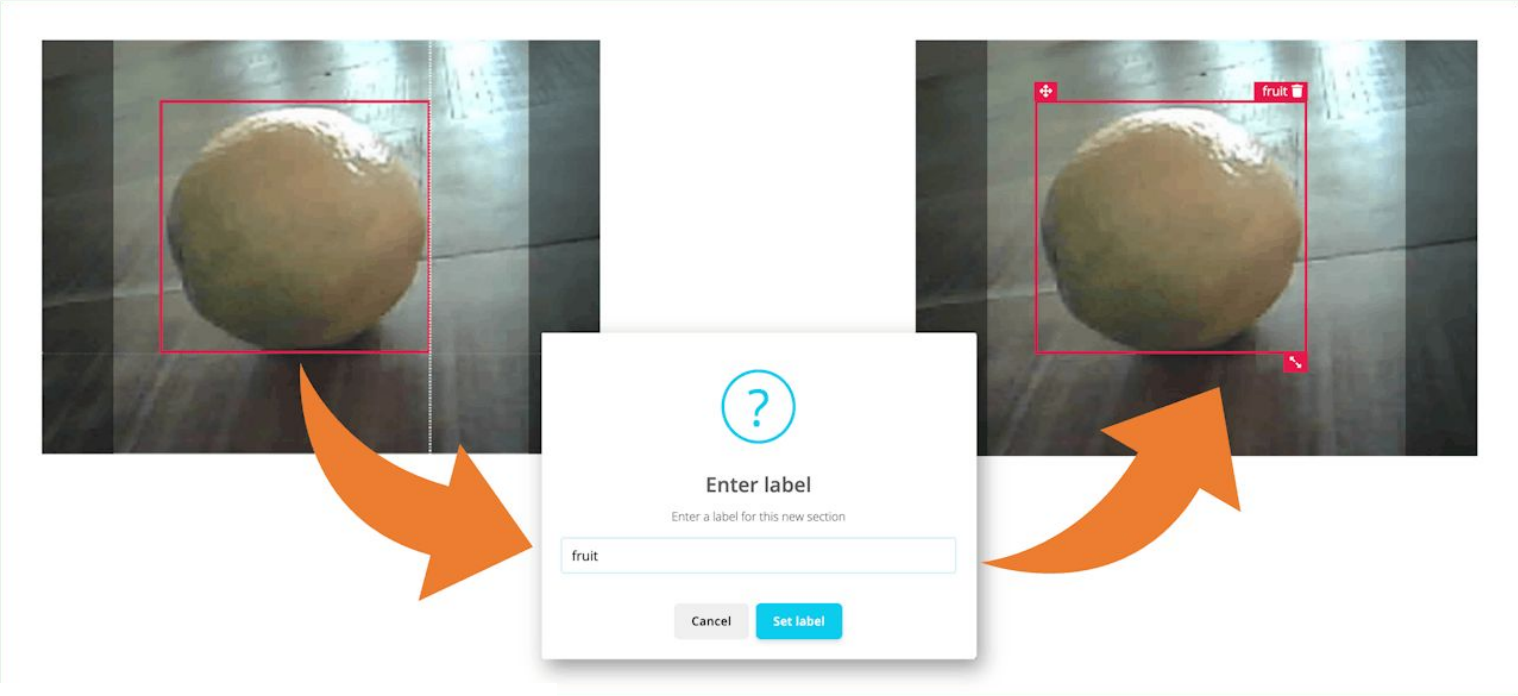
File > Examples > ESP32 > Camera > CameraWebServer.ino

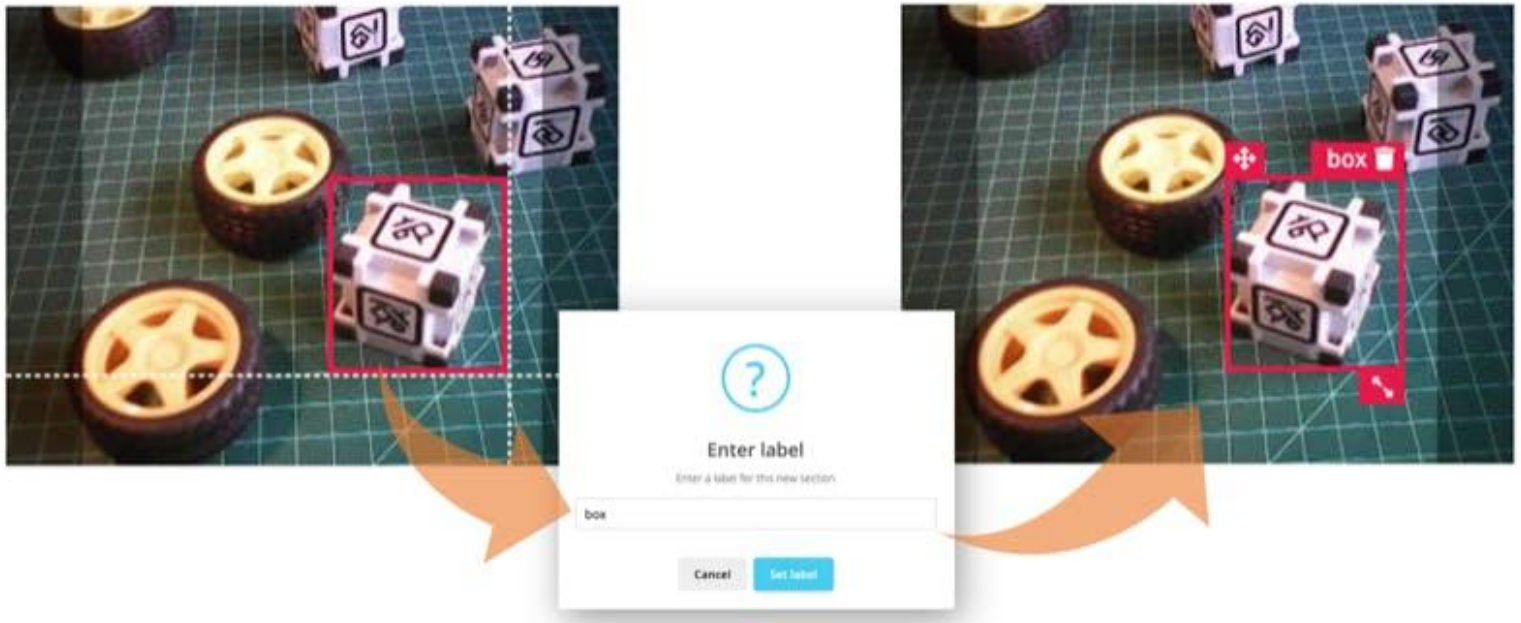


Labeling

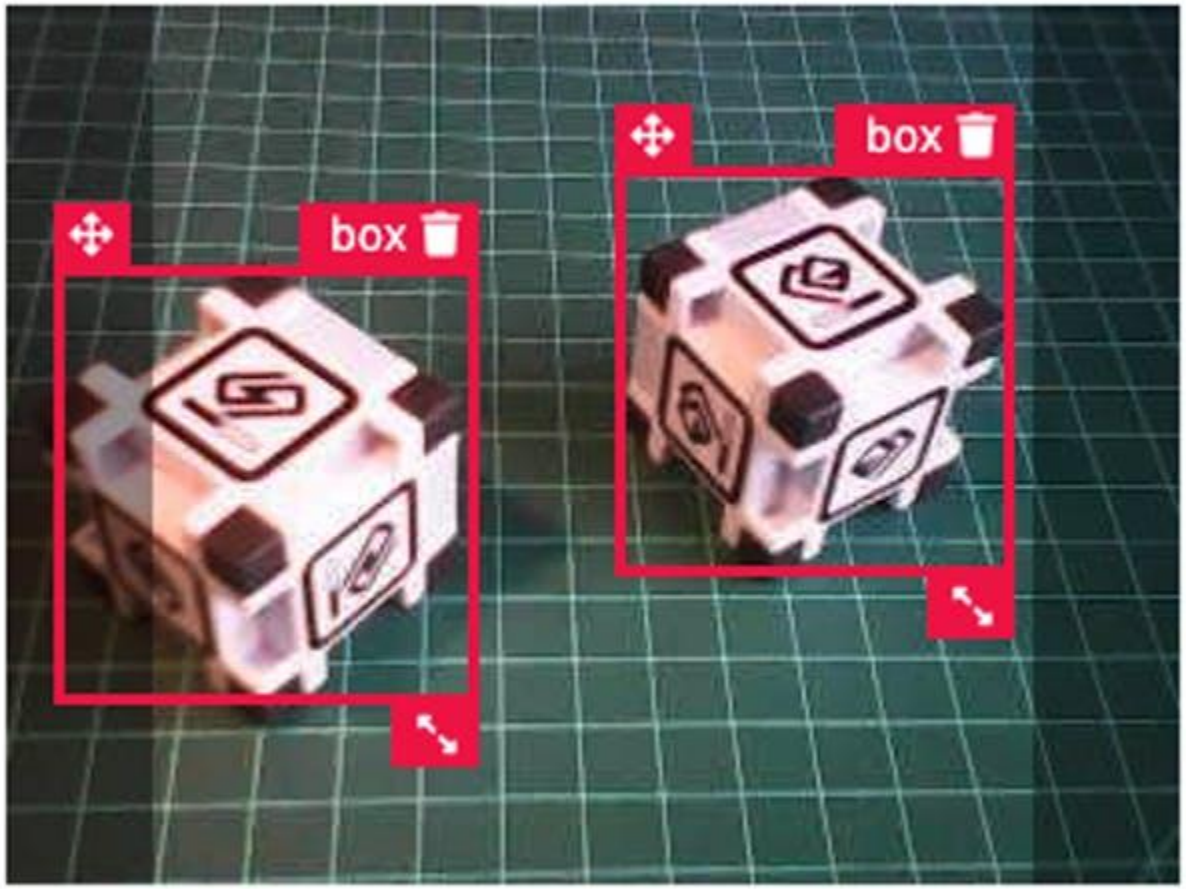

XIAO-ESP32S3-Sense-Objec
studio.edgeimpulse.com/studio/315759/acquisition/training?page=1
EDGE IMPULSE
MJRoBot (Marcelo Rovai) / XIAO-ESP32S3-Sense-Object_Detection
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
Dataset
Data sources
Labeling queue (47)
DATA COLLECTED
47 items
TRAIN / TEST SPLIT
-
Collect data
Connect a device to start building your dataset.
Dataset
Training (47)
Test (0)
SAMPLE NAME
LABELS
ADDED
LENGTH
20231128151645 - Today, 15:27:09 -
20231128150613 - Today, 15:27:09 -
20231128150604 - Today, 15:27:09 -
20231128150833 - Today, 15:27:09 -
20231128150600 - Today, 15:27:09 -
20231128150855 - Today, 15:27:09 -
20231128150458 - Today, 15:27:09 -
20231128150713 - Today, 15:27:09 -
20231128150908 - Today, 15:27:09 -
RAW DATA
20231128151645
Metadata
No metadata.

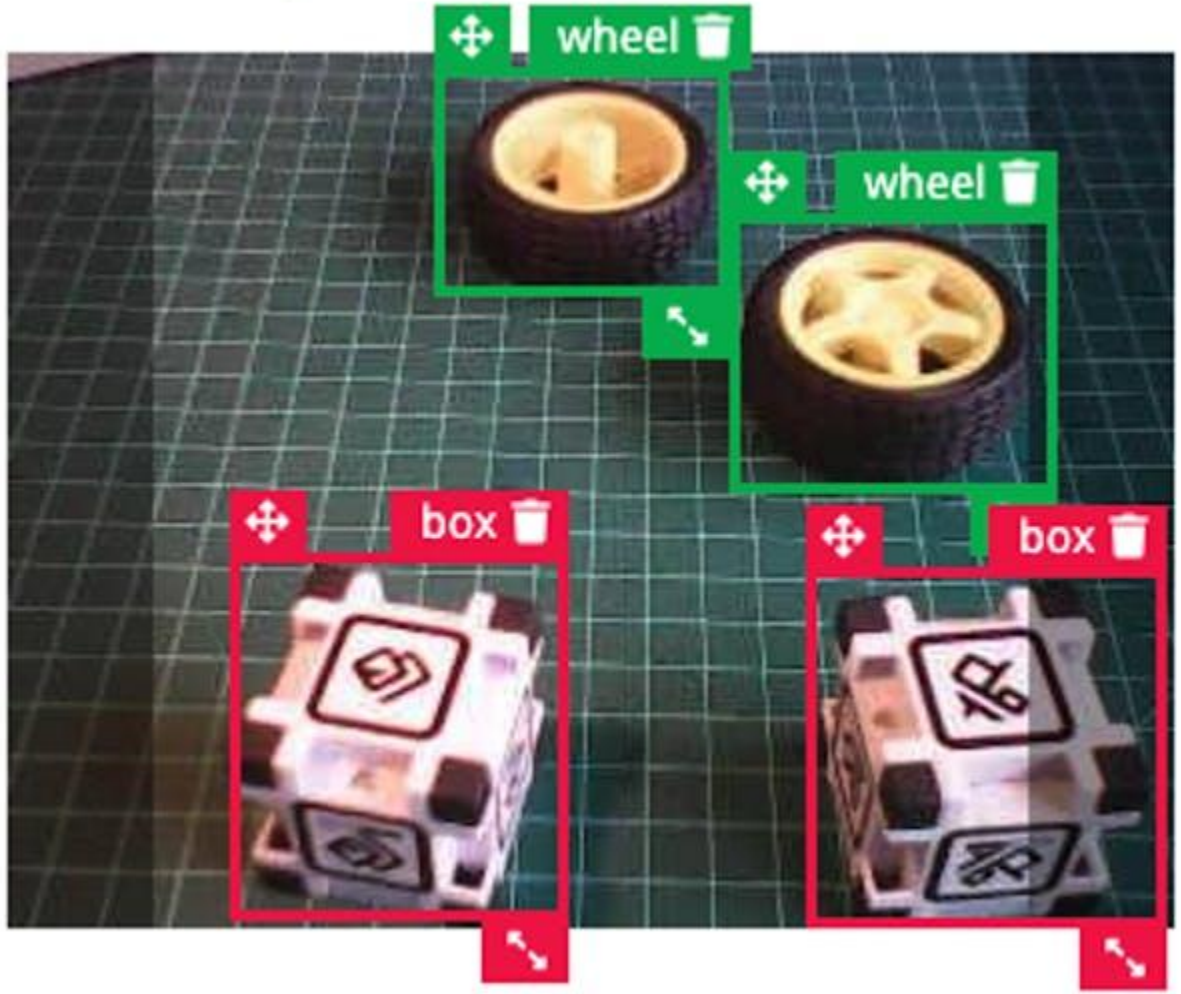
Labeling

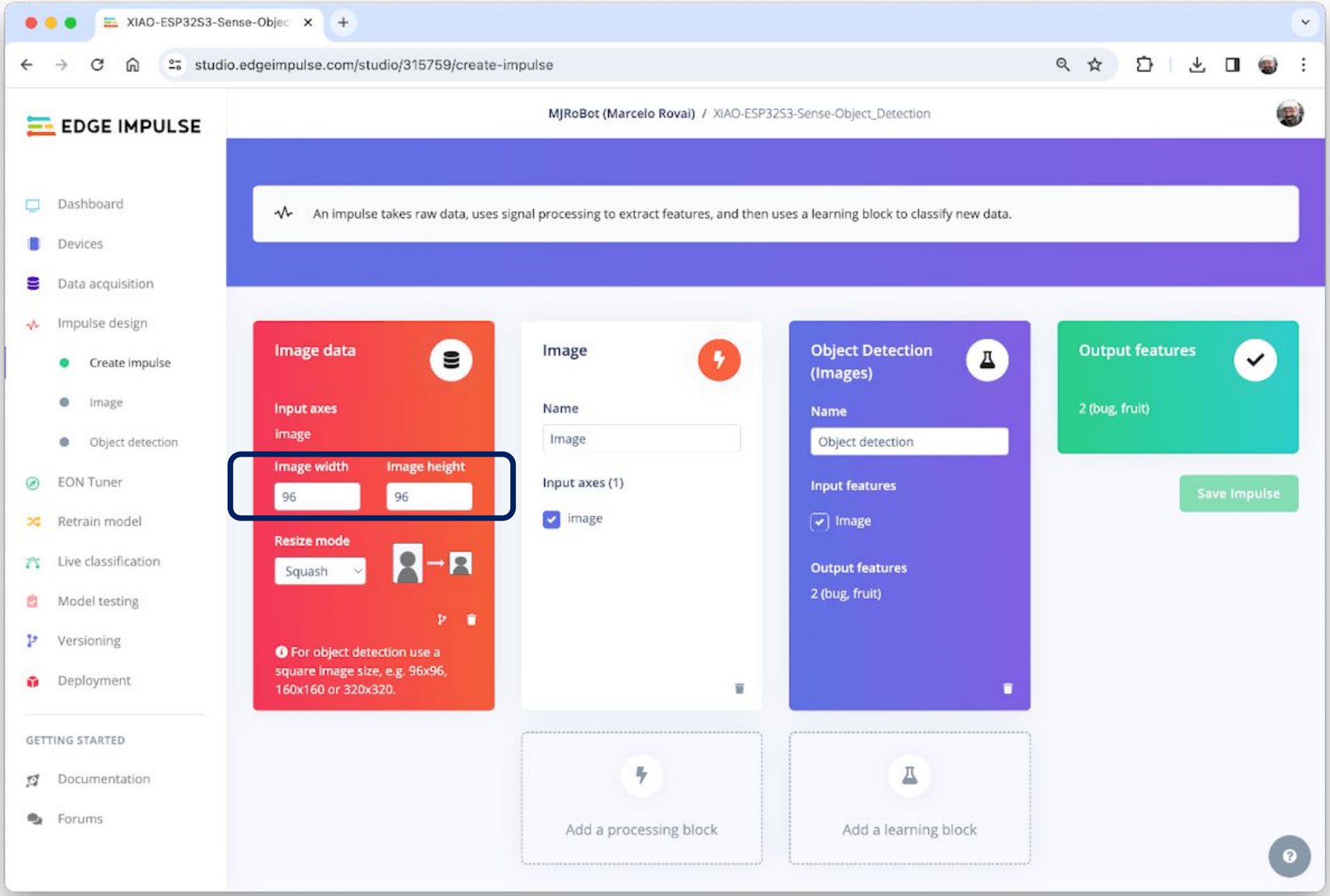

fruit
Enter label
Enter a label for this new section
fruit
Cancel
Set label


box
Enter label
box
Cancel
Set label

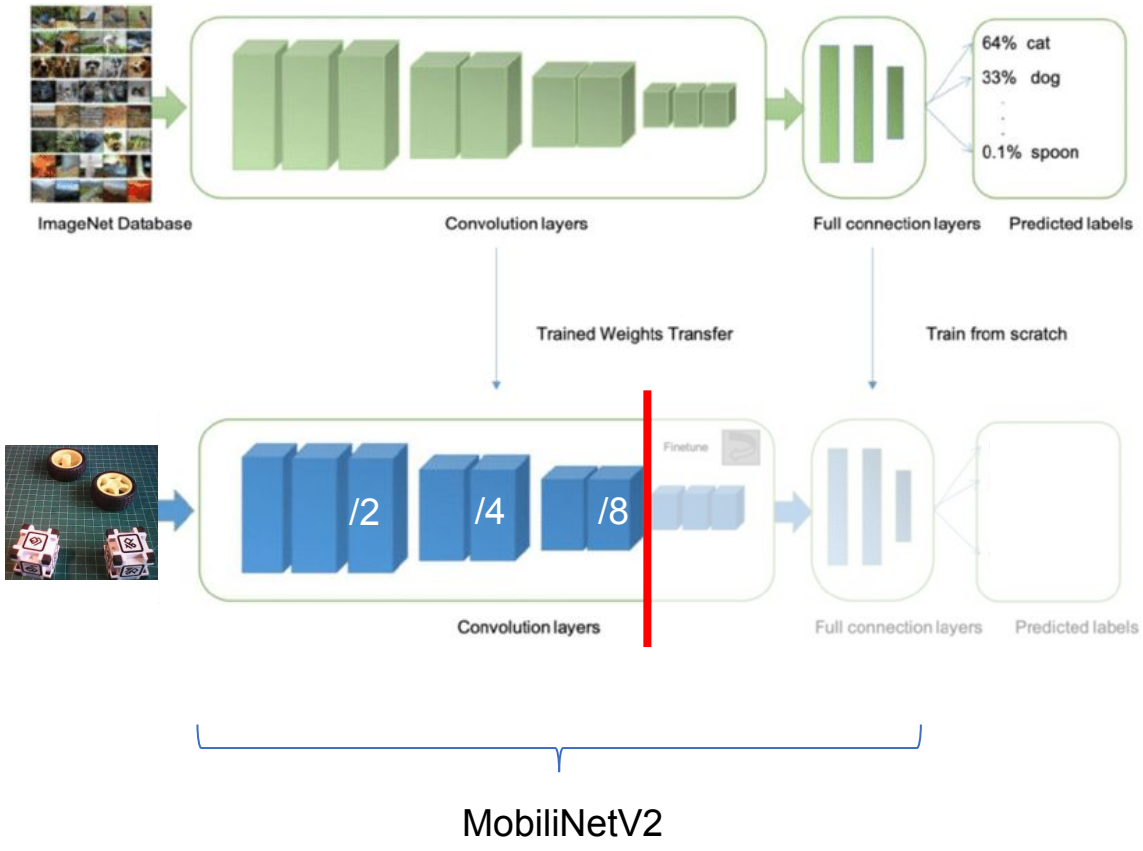
Labeling

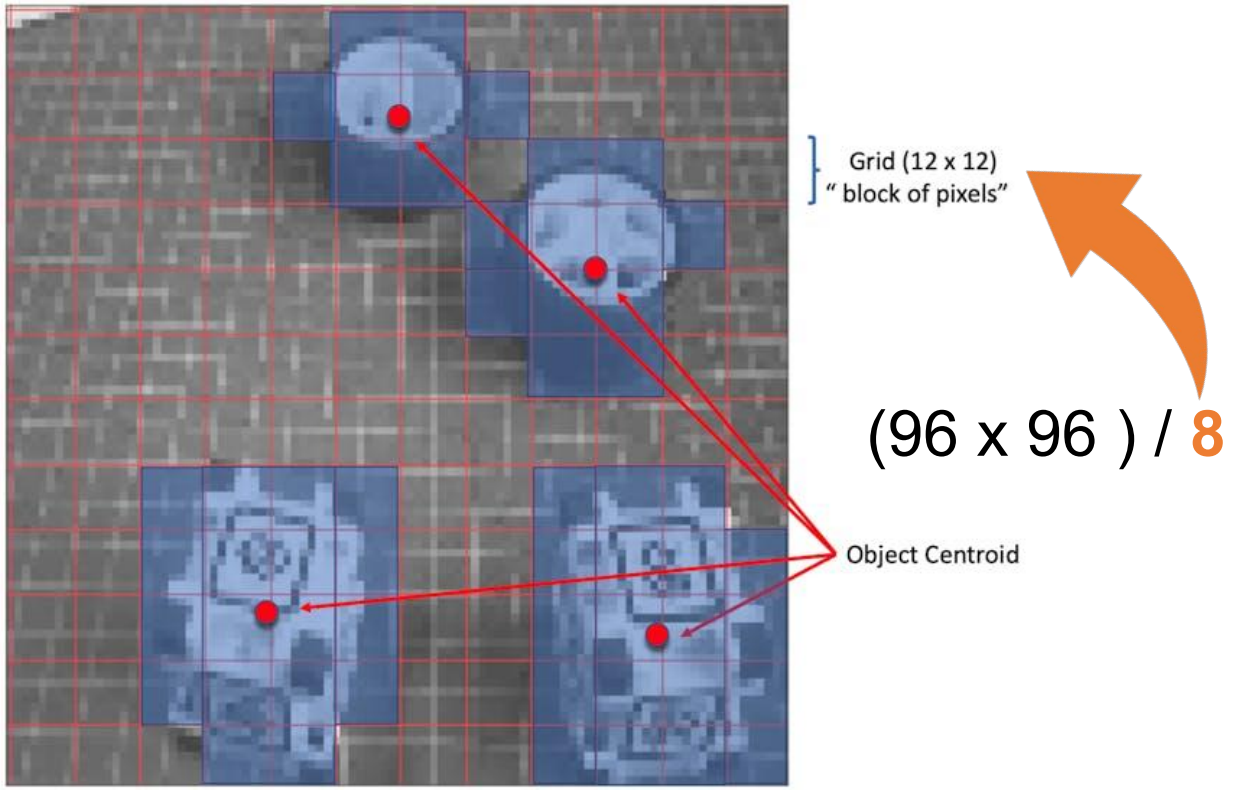




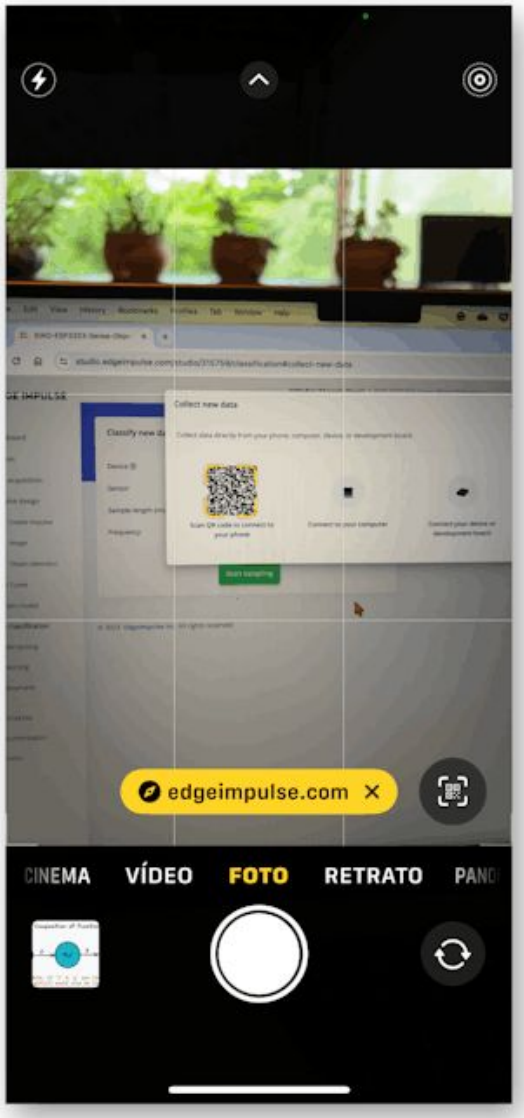

XIAO-ESP32S3-Sense-Objec
studio.edgeimpulse.com/studio/315759/create-impulse
EDGE IMPULSE
MJRoBot (Marcelo Rovai) / XIAO-ESP32S3-Sense-Object_Detection
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Object detection
EON Tuner
Retrain model
Live classification
Model testing
Versioning
Deployment
GETTING STARTED
Documentation
Forums
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 object detection use a square image size, e.g. 96x96, 160x160 or 320x320.
Image
Name
Image
Input axes (1)
image
Object Detection (Images)
Name
Object detection
Input features
Image
Output features
2 (bug, fruit)
Output features
2 (bug, fruit)
Save Impulse
Add a processing block
Add a learning block

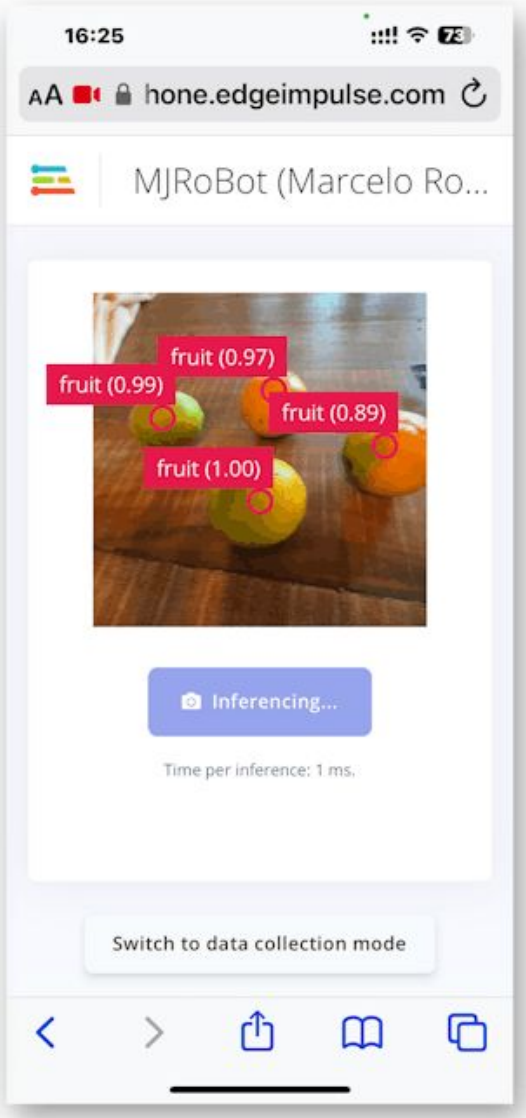
Model: FOMO

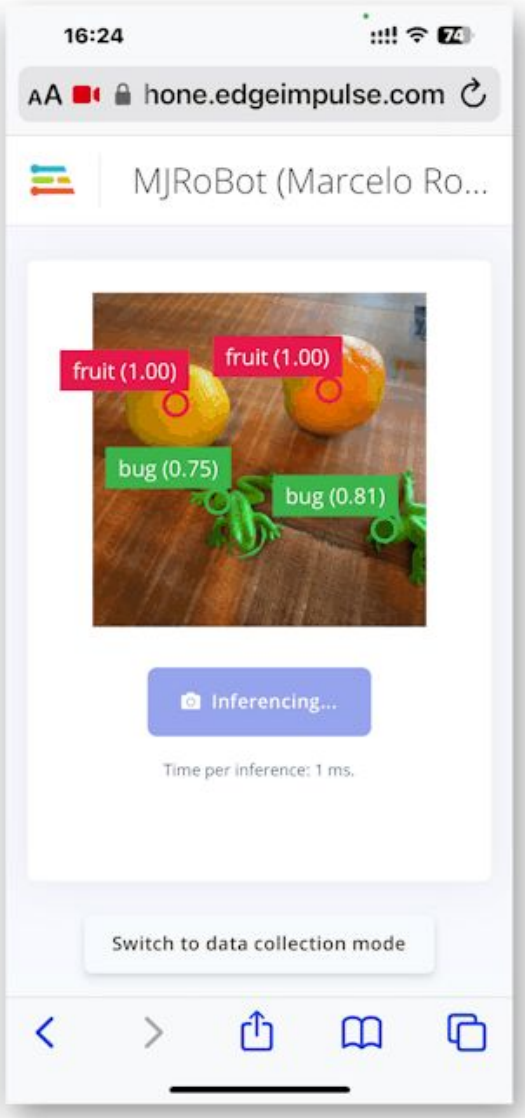

ImageNet Database
Convolution layers
Full connection layers
Predicted labels
64% cat
33% dog
0.1% spoon
Trained Weights Transfer
Train from scratch
/2
/4
/8
Finetune
Convolution layers
Full connection layers
Predicted labels
MobiliNetV2

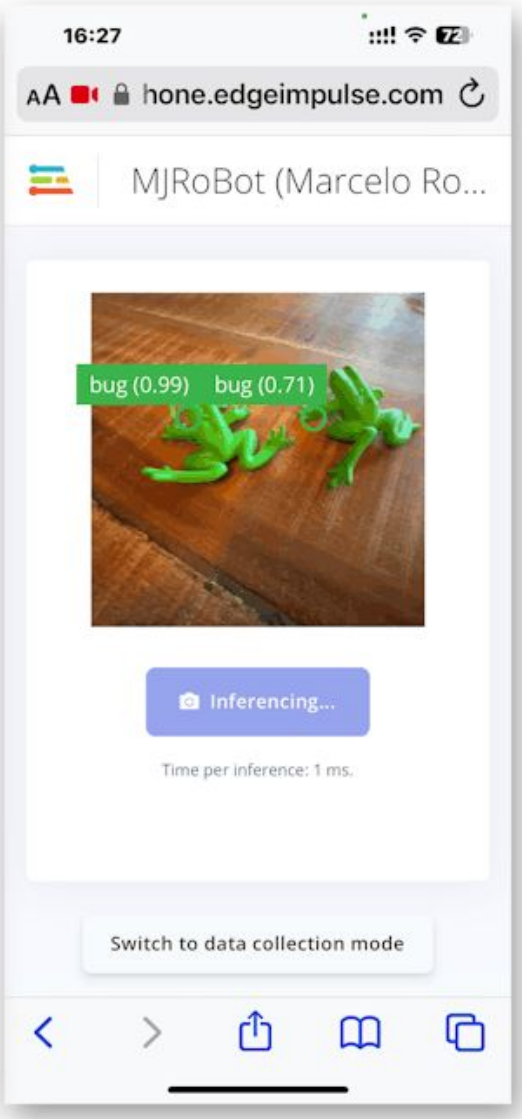

Grid (12 x 12)
" block of pixels"
(96 x 96) / 8
Object Centroid

Inference Test





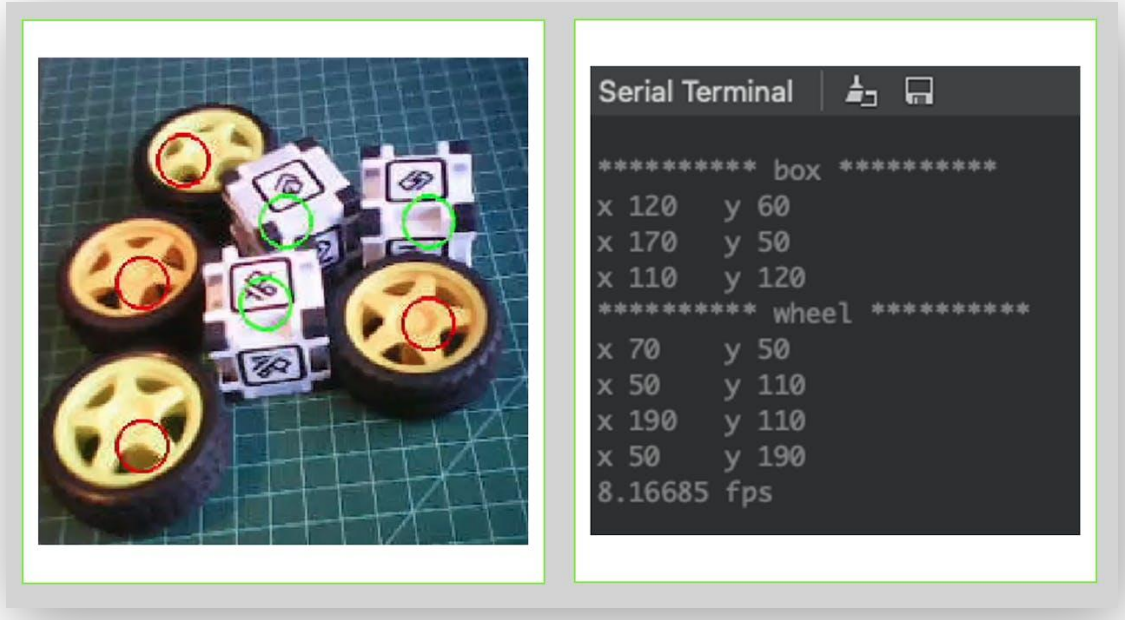


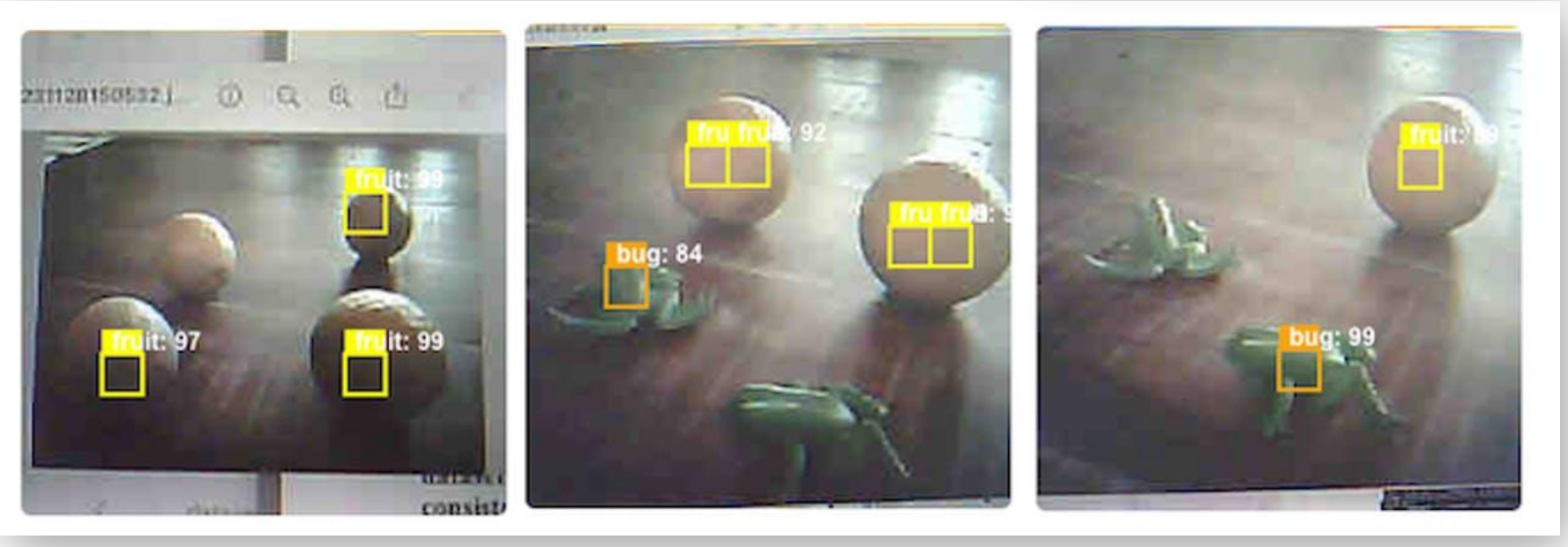


Deploy

OpenMV

(Nicla)

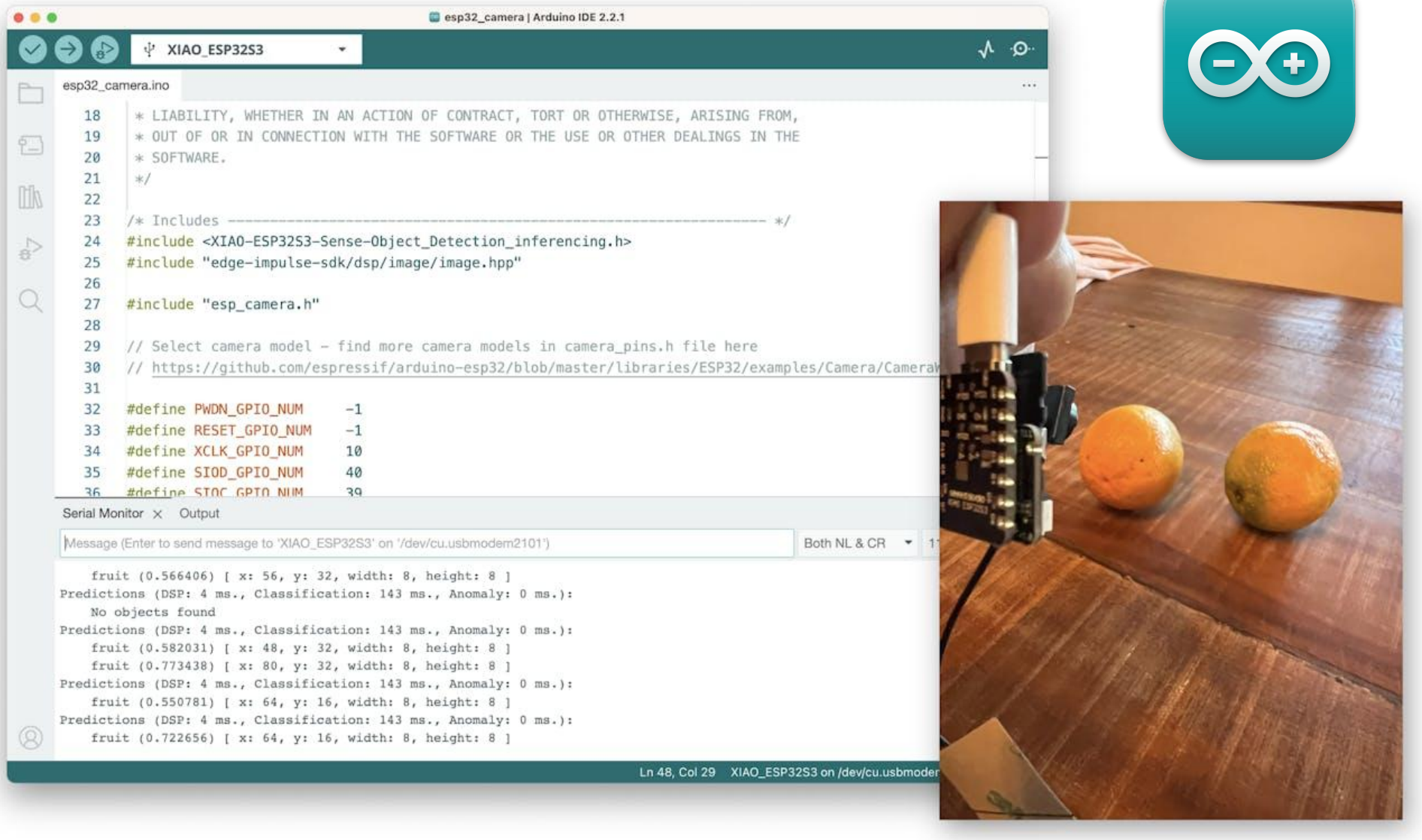




SenseCraft

(XIAO)

Deploy


esp32_camera | Arduino IDE 2.2.1
XIAO_ESP32S3
esp32_camera.ino
18 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
19 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
20 * SOFTWARE.
21 */
22
23 /* Includes ---------- */
24 #include <XIAO-ESP32S3-Sense-Object_Detection_inferencing.h>
25 #include "edge-impulse-sdk/dsp/image/image.hpp"
26
27 #include "esp_camera.h"
28
29 // Select camera model - find more camera models in camera_pins.h file here
30 // https://github.com/espressif/arduino-esp32/blob/master/libraries/ESP32/examples/Camera/CameraW
31
32 #define PWDN_GPIO_NUM -1
33 #define RESET_GPIO_NUM -1
34 #define XCLK_GPIO_NUM 10
35 #define SIOD_GPIO_NUM 40
36 #define SIOC_GPIO_NUM 39
Serial Monitor
Output
Message (Enter to send message to 'XIAO_ESP32S3' on '/dev/cu.usbmodem2101')
Both NL & CR
fruit (0.566406) [x: 56, y: 32, width: 8, height: 8]
Predictions (DSP: 4 ms., Classification: 143 ms., Anomaly: 0 ms.):
No objects found
Predictions (DSP: 4 ms., Classification: 143 ms., Anomaly: 0 ms.):
fruit (0.582031) [x: 48, y: 32, width: 8, height: 8]
fruit (0.773438) [x: 80, y: 32, width: 8, height: 8]
Predictions (DSP: 4 ms., Classification: 143 ms., Anomaly: 0 ms.):
fruit (0.550781) [x: 64, y: 16, width: 8, height: 8]
Predictions (DSP: 4 ms., Classification: 143 ms., Anomaly: 0 ms.):
fruit (0.722656) [x: 64, y: 16, width: 8, height: 8]
Ln 48, Col 29
XIAO_ESP32S3 on /dev/cu.usbmoder

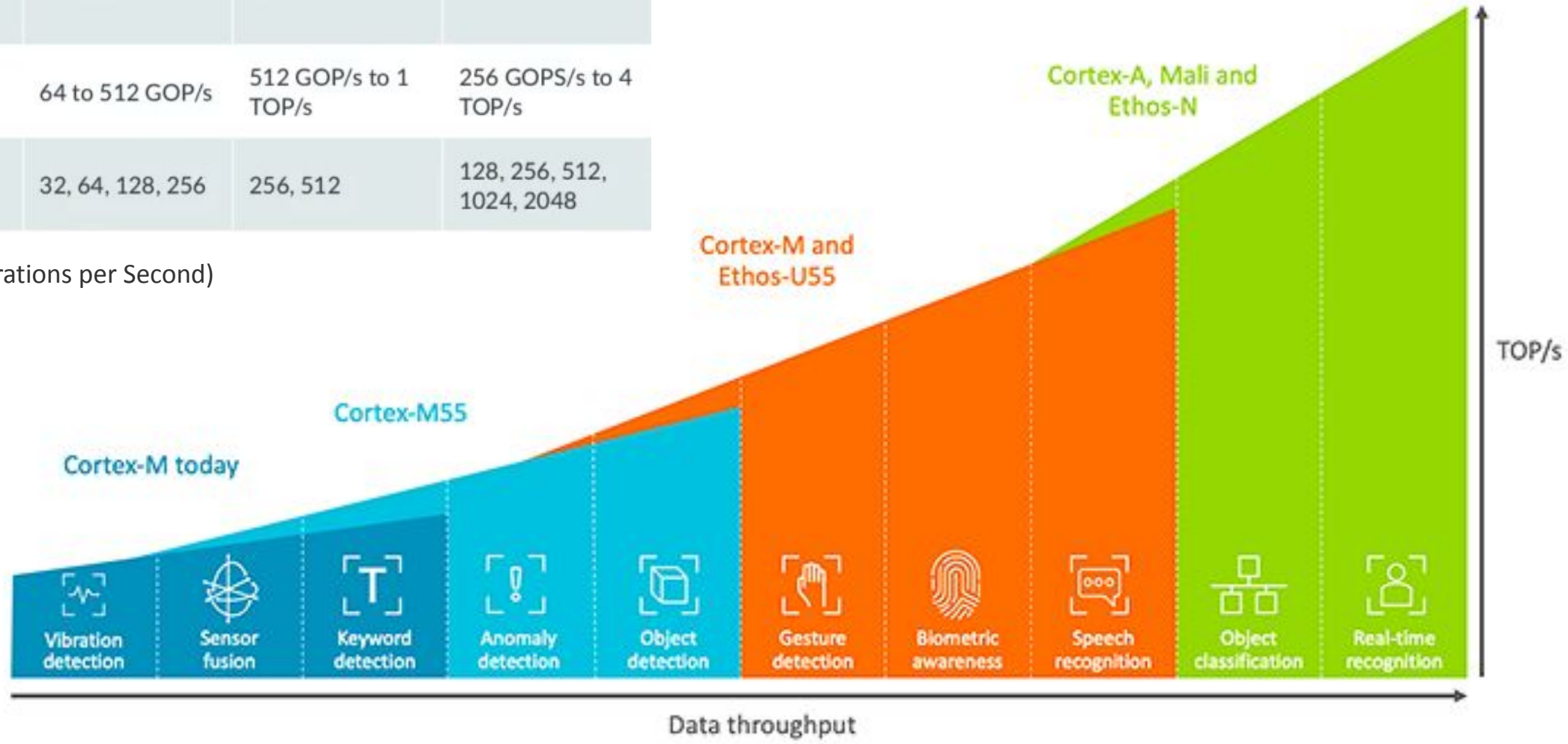
microNPU

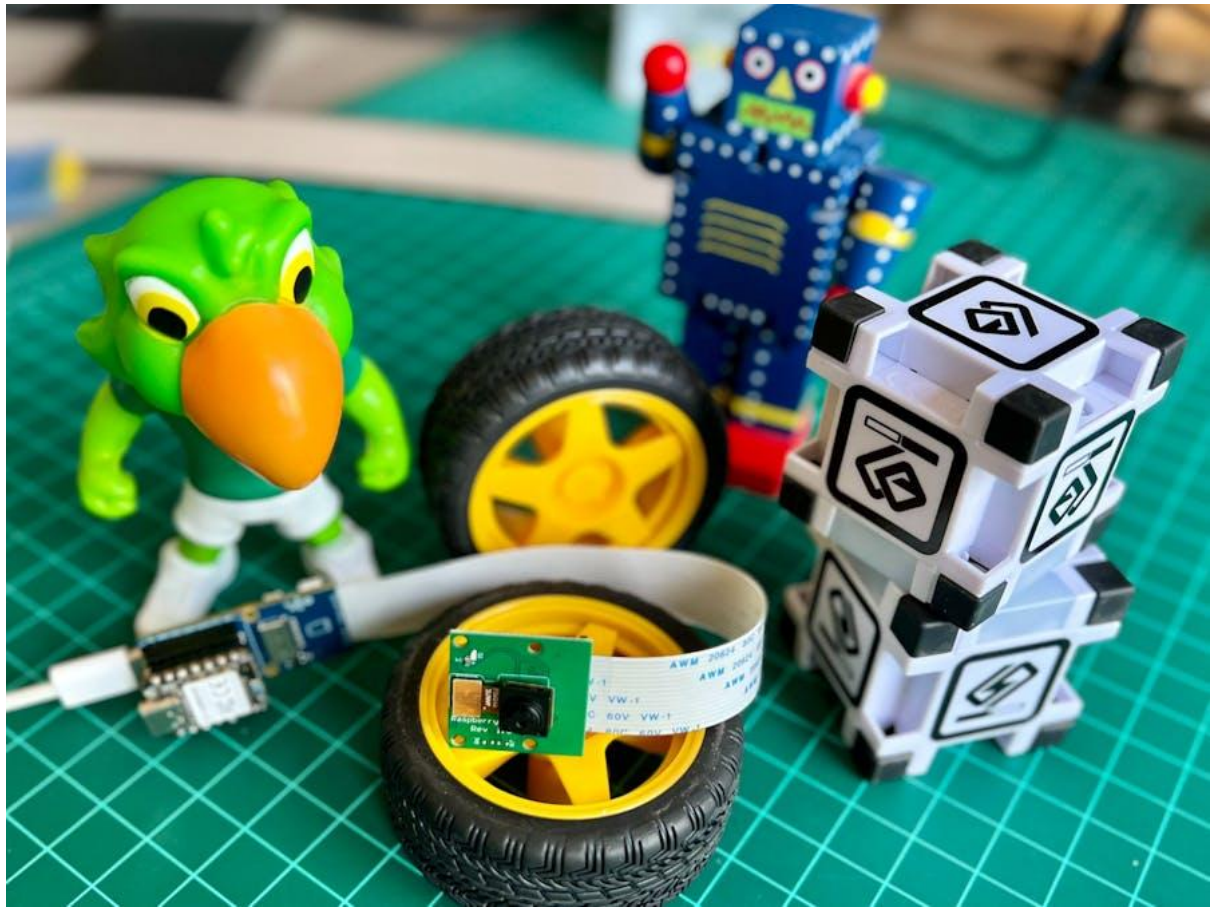
a neural network unit for TinyML

ML- optimized Solutions

	Ethos-U55	Ethos-U65	Ethos-U85
Performance (At 1 GHz)	64 to 512 GOP/s	512 GOP/s to 1 TOP/s	256 GOPS/s to 4 TOP/s
MACs (8x8)	32, 64, 128, 256	256, 512	128, 256, 512, 1024, 2048

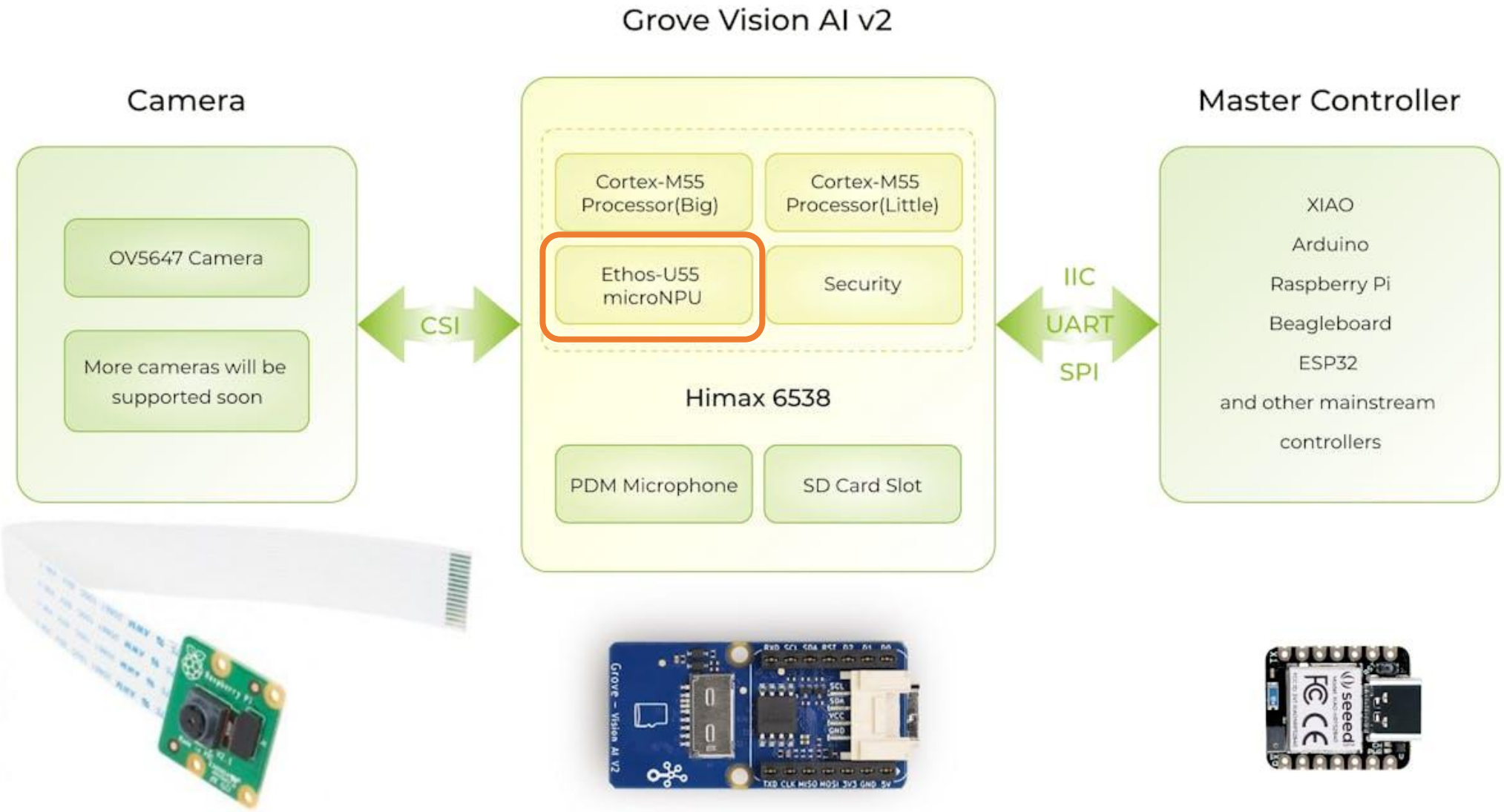
TOPS (Tera Operations per Second)





Computer Vision at the Edge with Grove Vision AI Module V2

Exploring Computer Vision applications such as Image Classification, Object Detection, and Pose estimation.

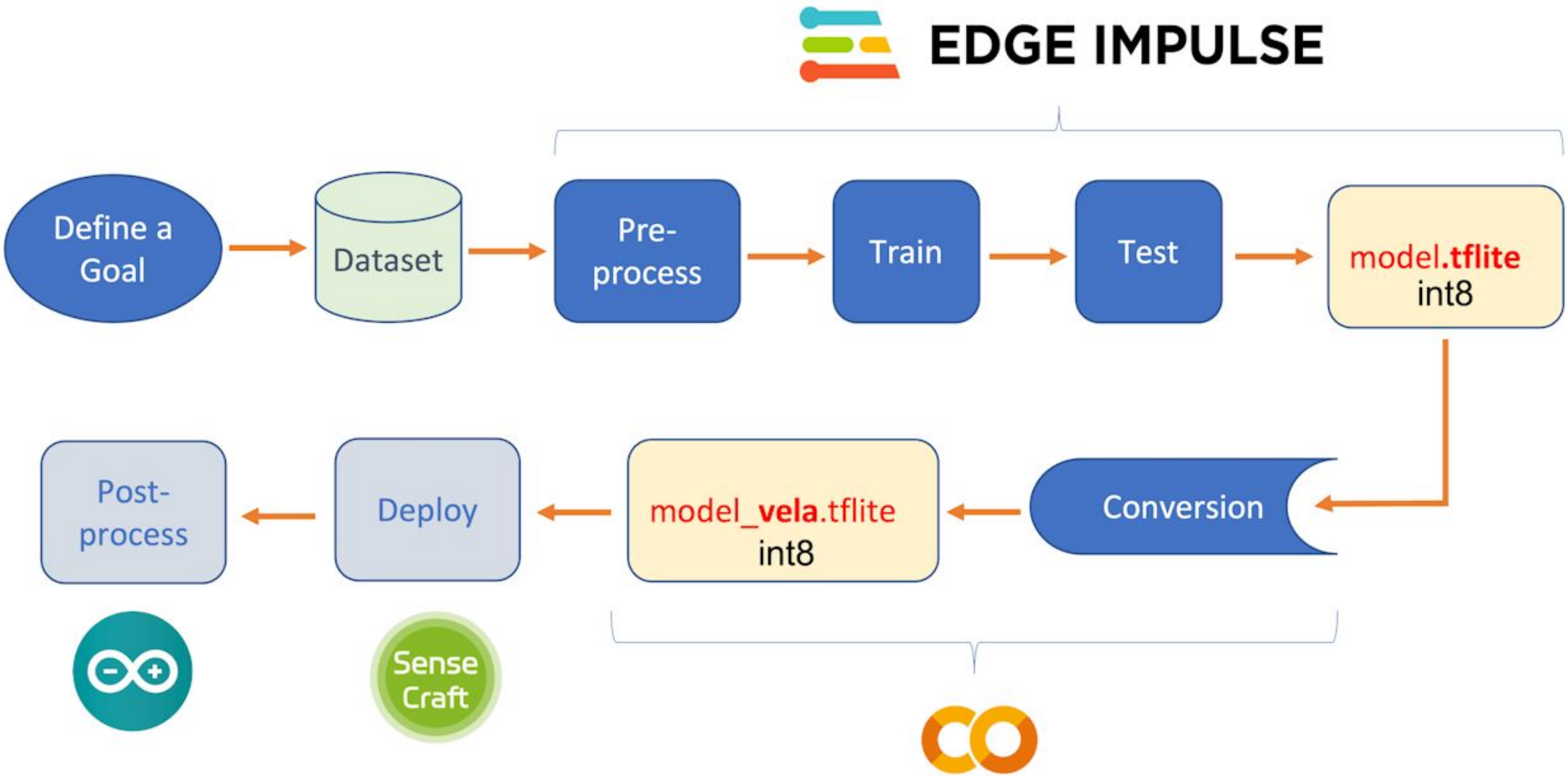

Grove Vision AI v2
Camera
OV5647 Camera
More cameras will be supported soon
CSI
Cortex-M55 Processor(Big)
Cortex-M55 Processor(Little)
Ethos-U55 microNPU
Security
Himax 6538
PDM Microphone
SD Card Slot
IIC
UART
SPI
Master Controller
XIAO
Arduino
Raspberry Pi
Beagleboard
ESP32
and other mainstream controllers

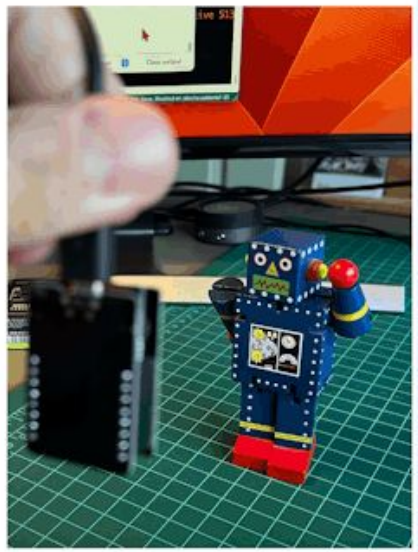
Person Classification

Face Detection

Person Detection

Human Pose Detection


EDGE IMPULSE
Define a Goal
Dataset
Pre-process
Train
Test
model.tflite int8
Conversion
model_vela.tflite int8
Deploy
Post-process
Sense Craft



Classification: 687 ms

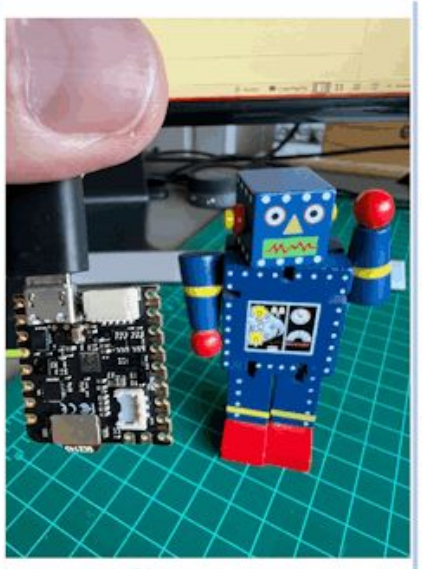
1.5 FPS

ESP - CAM
Xtensa LX6
240 MHz

Classification: 142 ms

7.0 FPS

XIAO ESP32S3
Xtensa LX7
240 MHz



Classification: 86 ms

11.6 FPS

Nicla-Vision
ARM M7
480 MHz

Classification: 83 ms

12.0 FPS

Portenta H7
ARM M7
480 MHz

Classification: 6 ms

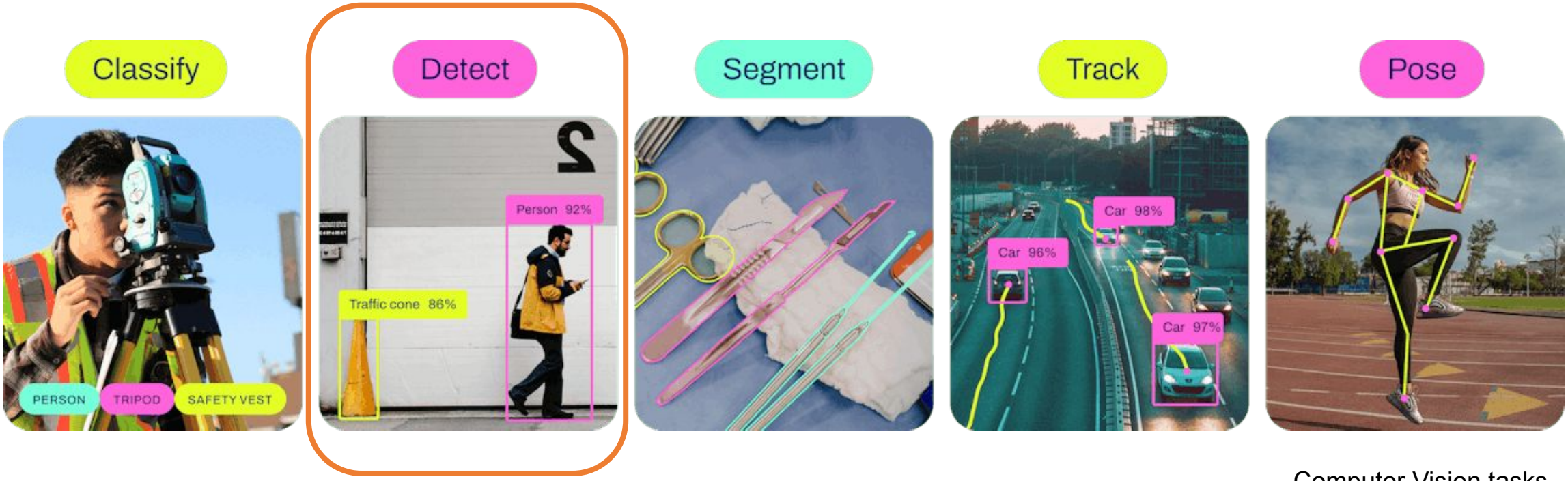
167 FPS

Grove Vision AI V2
ARM Ethus-U55
400 MHz

YOLO

Object Detection model

Ultralytics YOLO (You Only Look Once)



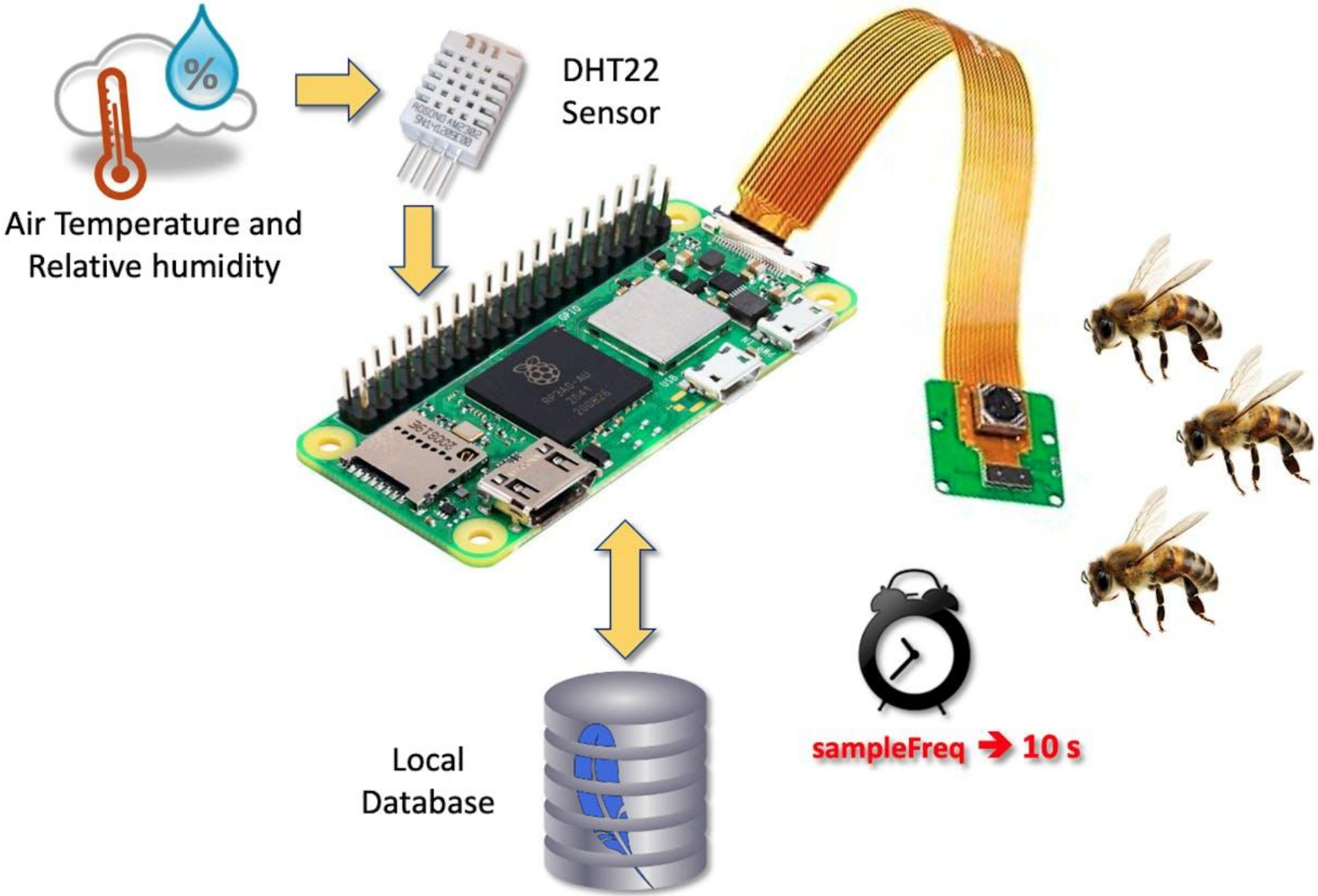
Computer Vision tasks

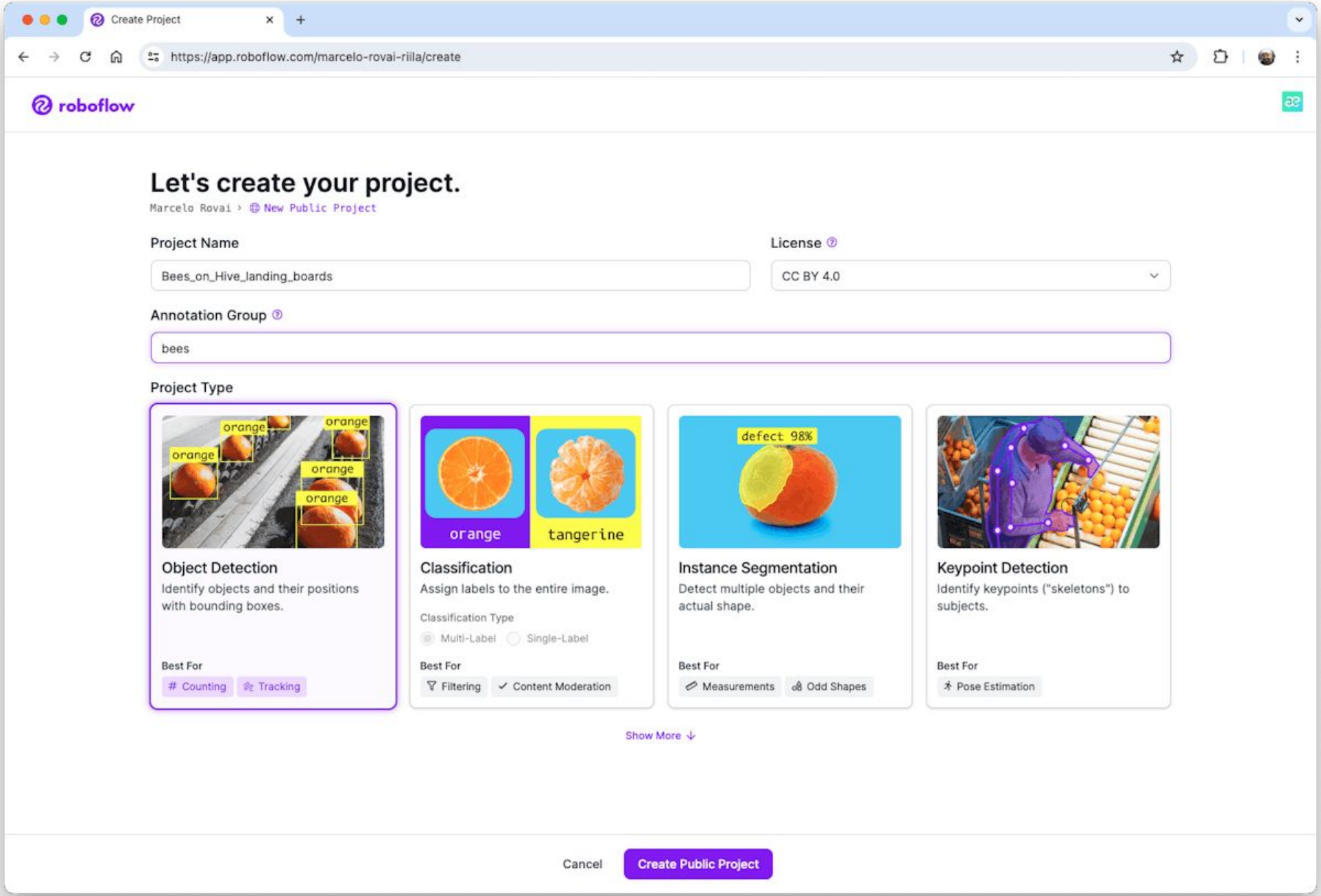
Real-time **object detection** systems that identify and classify many objects **very fast** in a single image pass.

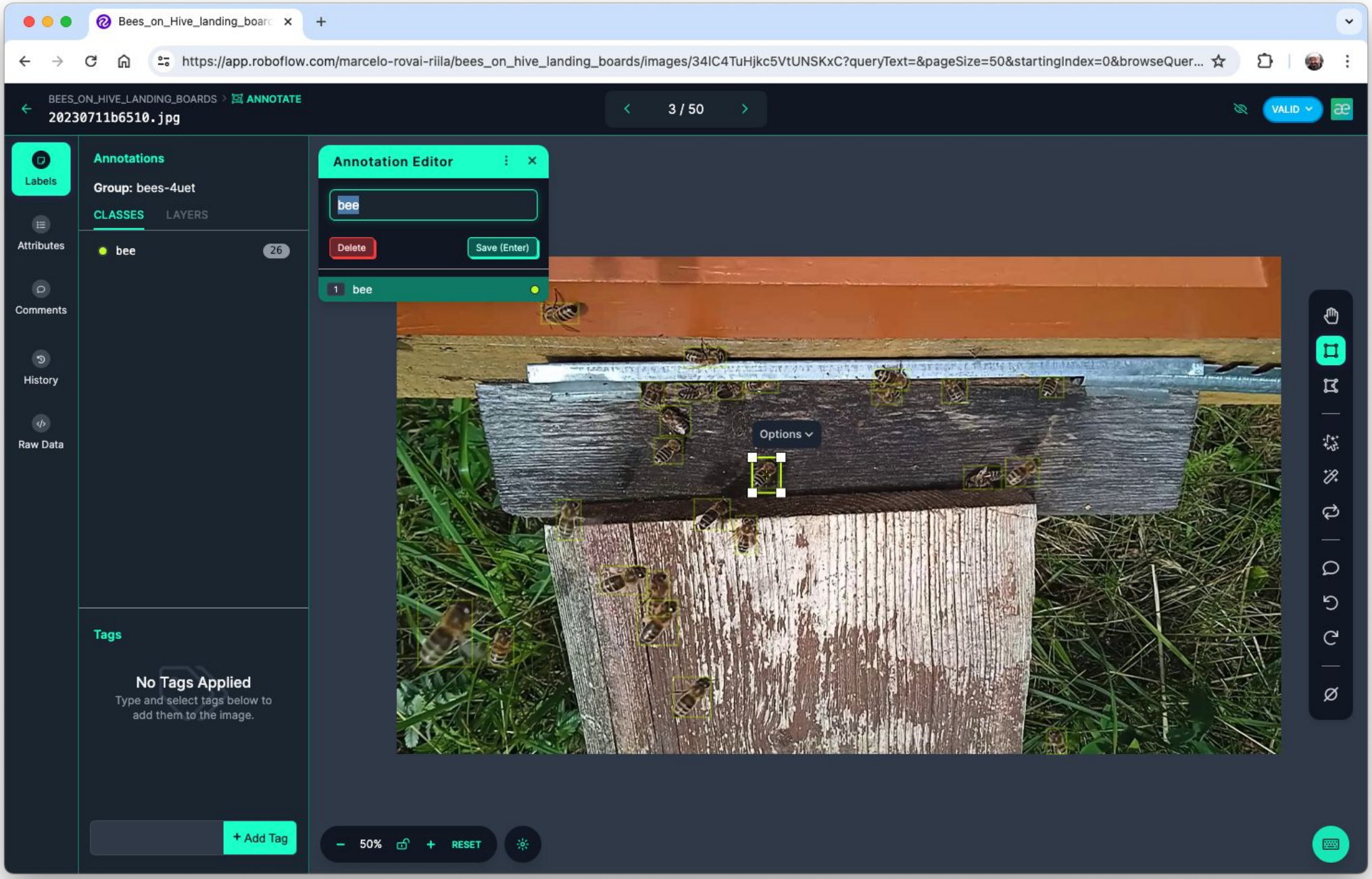
BuzzTech: Machine Learning at the Edge

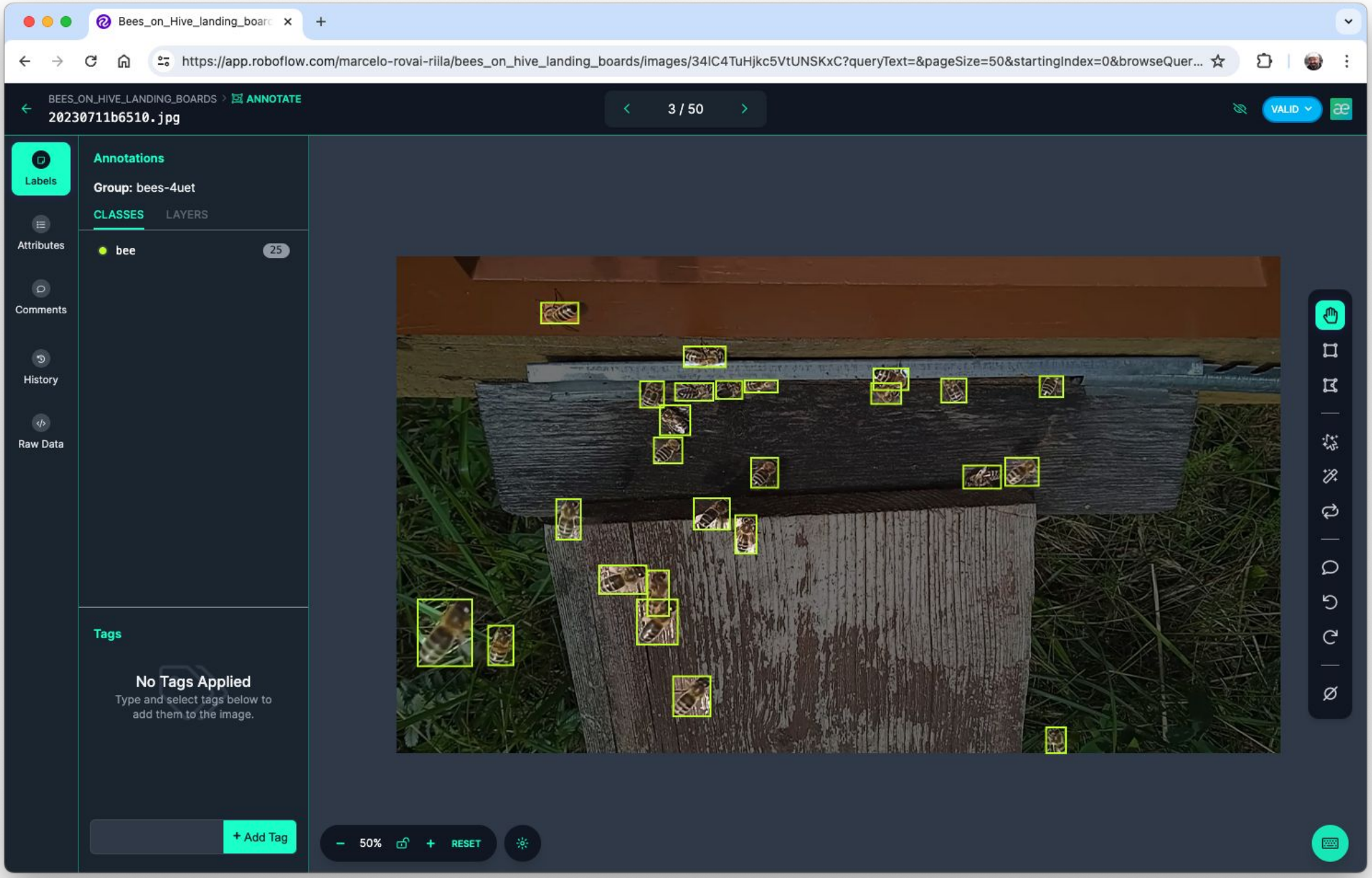
Deploying YOLOv8 on Raspberry Pi Zero 2W for Real-Time Bee Counting at the Hive Entrance.

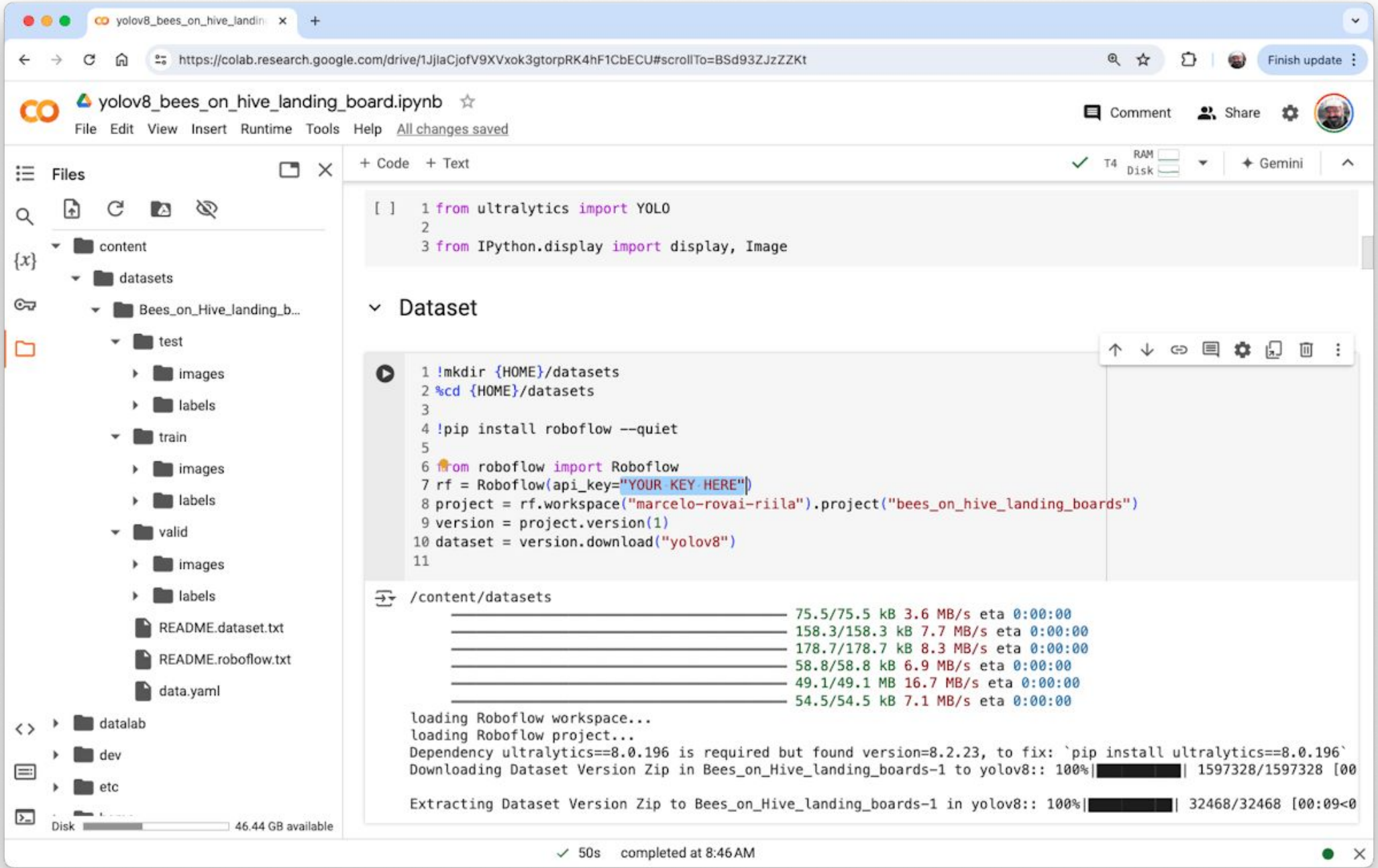
Goal: Estimate the number of bees


Air Temperature and
Relative humidity
DHT22
Sensor
Local
Database
sampleFreq → 10 s


Create Project
https://app.roboflow.com/marcelo-rovai-riila/create
roboflow
Let's create your project.
Marcelo Rovai › New Public Project
Project Name
Bees_on_Hive_landing_boards
License
CC BY 4.0
Annotation Group
bees
Project Type
orange
orange
orange
orange
orange
Object Detection
Identify objects and their positions with bounding boxes.
Best For
Counting
Tracking
orange
tangerine
Classification
Assign labels to the entire image.
Classification Type
Multi-Label
Single-Label
Best For
Filtering
Content Moderation
defect 98%
Instance Segmentation
Detect multiple objects and their actual shape.
Best For
Measurements
Odd Shapes
Keypoint Detection
Identify keypoints ("skeletons") to subjects.
Best For
Pose Estimation
Show More
Cancel
Create Public Project


Bees_on_Hive_landing_boarc
https://app.roboflow.com/marcelo-rovai-riila/bees_on_hive_landing_boards/images/34lC4TuHjkc5VtUNSKxC?queryText=&pageSize=50&startingIndex=0&browseQuer...
BEES_ON_HIVE_LANDING_BOARDS
ANNOTATE
20230711b6510.jpg
3 / 50
VALID
Labels
Attributes
Comments
History
Raw Data
Annotations
Group: bees-4uet
CLASSES
LAYERS
bee
26
Tags
No Tags Applied
Type and select tags below to add them to the image.
+ Add Tag
Annotation Editor
bee
Delete
Save (Enter)
1 bee
Options
50%
RESET


Bees_on_Hive_landing_boarc
https://app.roboflow.com/marcelo-rovai-riila/bees_on_hive_landing_boards/images/34lC4TuHjkc5VtUNSKxC?queryText=&pageSize=50&startingIndex=0&browseQuer...
BEES_ON_HIVE_LANDING_BOARDS
ANNOTATE
20230711b6510.jpg
3 / 50
VALID
Labels
Attributes
Comments
History
Raw Data
Annotations
Group: bees-4uet
CLASSES
LAYERS
bee
25
Tags
No Tags Applied
Type and select tags below to add them to the image.
+ Add Tag
50%
RESET

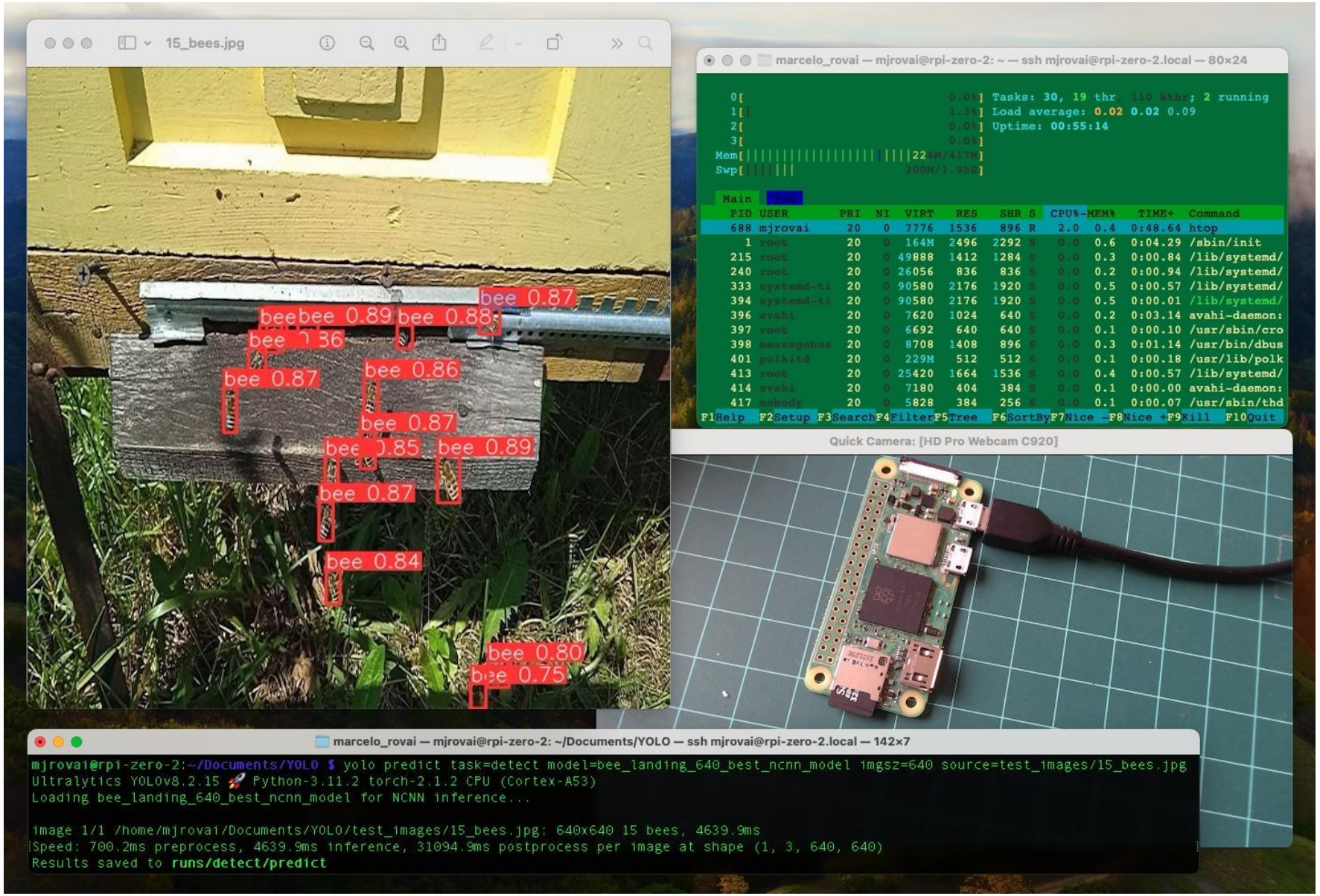

yolov8_bees_on_hive_landin
https://colab.research.google.com/drive/1JjlaCjofV9XVxok3gtorpRK4hF1CbECU#scrollTo=BSd93ZJzZZKt
Finish update
yolov8_bees_on_hive_landing_board.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
Files
+ Code + Text
T4
RAM
Disk
Gemini
content
datasets
Bees_on_Hive_landing_b...
test
images
labels
train
images
labels
valid
images
labels
README.dataset.txt
README.roboflow.txt
data.yaml
datalab
dev
etc
Disk 46.44 GB available
[ ] 1 from ultralytics import YOLO
2
3 from IPython.display import display, Image
Dataset
1 !mkdir {HOME}/datasets
2 %cd {HOME}/datasets
3
4 !pip install roboflow --quiet
5
6 from roboflow import Roboflow
7 rf = Roboflow(api_key="YOUR KEY HERE")
8 project = rf.workspace("marcelo-rovai-riila").project("bees_on_hive_landing_boards")
9 version = project.version(1)
10 dataset = version.download("yolov8")
11
/content/datasets
75.5/75.5 kB 3.6 MB/s eta 0:00:00
158.3/158.3 kB 7.7 MB/s eta 0:00:00
178.7/178.7 kB 8.3 MB/s eta 0:00:00
58.8/58.8 kB 6.9 MB/s eta 0:00:00
49.1/49.1 MB 16.7 MB/s eta 0:00:00
54.5/54.5 kB 7.1 MB/s eta 0:00:00
loading Roboflow workspace...
loading Roboflow project...
Dependency ultralytics==8.0.196 is required but found version=8.2.23, to fix: `pip install ultralytics==8.0.196`
Downloading Dataset Version Zip in Bees_on_Hive_landing_boards-1 to yolov8:: 100%| 1597328/1597328 [00
Extracting Dataset Version Zip to Bees_on_Hive_landing_boards-1 in yolov8:: 100%| 32468/32468 [00:09<0
50s completed at 8:46 AM

Raw Dataset

Dataset Labeling (Roboflow)

Model Training **Ultralytics** (PyTorch)

Model Inference **Ultralytics - NCNN** (Rasp)



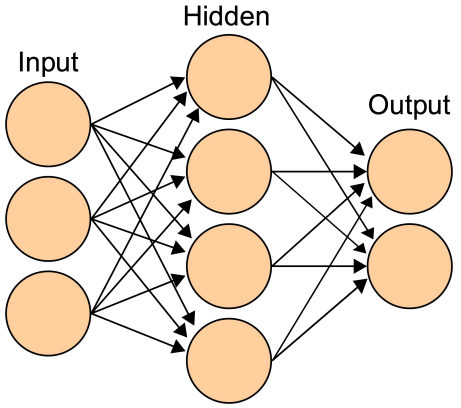
LSTM

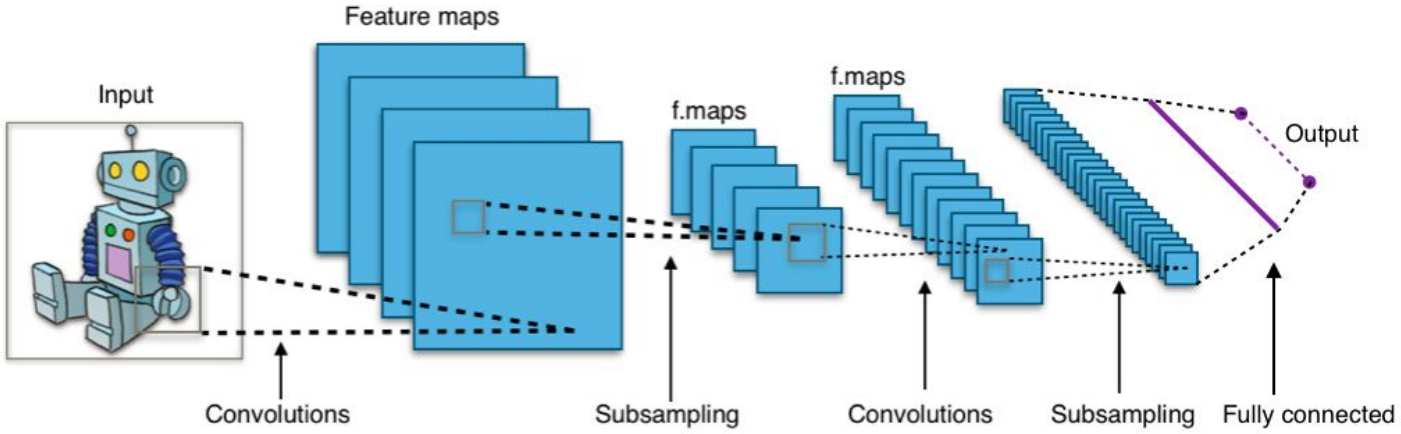
Long Short-Term Model

LSTM model is a type of recurrent neural network (RNN) that is well-suited for ***sequence prediction*** *problems by effectively capturing long-term dependencies in data sequences.*

Deep Learning models (or artificial neural networks)

Fully Connected Neural Networks (FCNNs): Networks where **each neuron in one layer is connected to every neuron in the following layer**, useful for complex pattern recognition across diverse datasets.

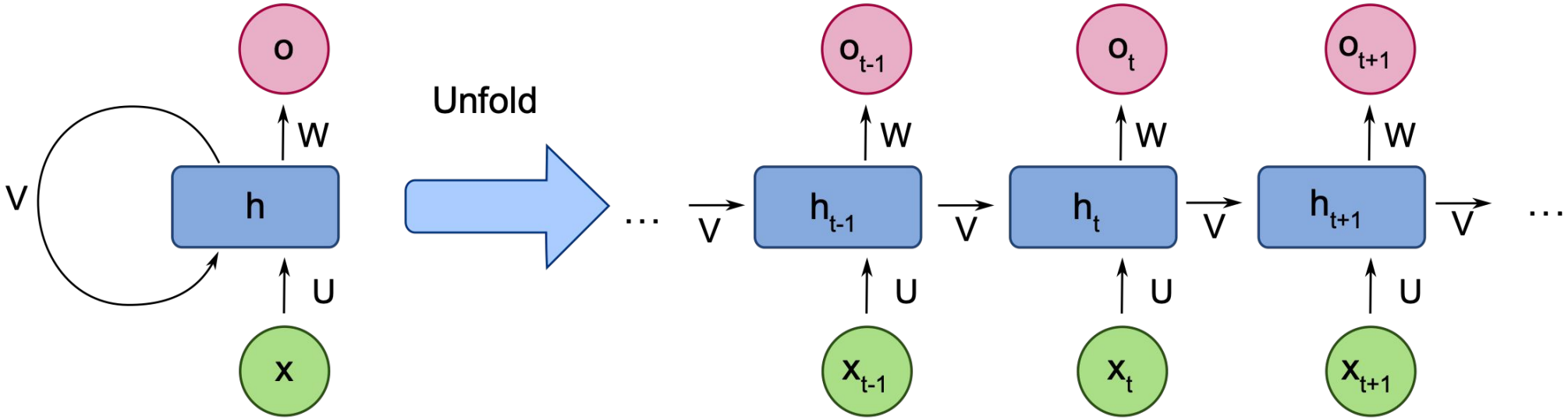




Convolutional Neural Networks (CNNs): Specialized for **grid-like data such as images**, using convolutional layers to detect and learn spatial hierarchies of features.

Deep Learning models (or artificial neural networks)

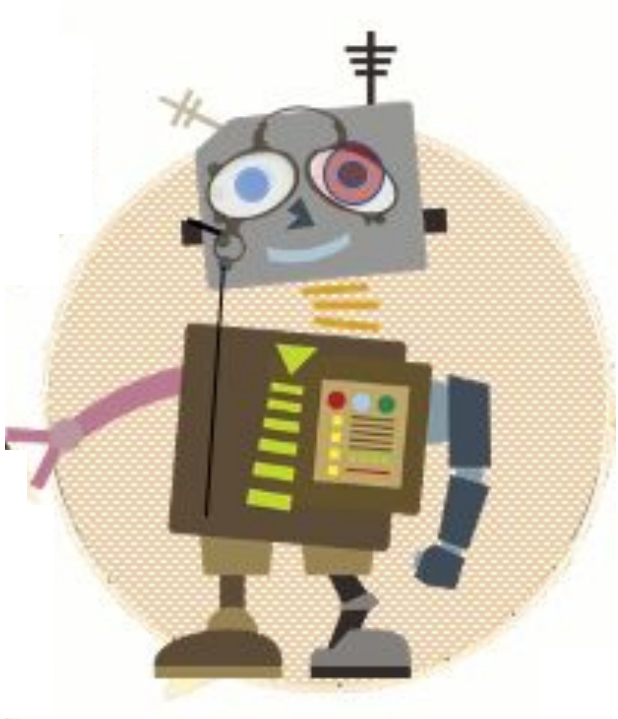
Recurrent Neural Networks (RNNs): Designed for **sequential data like time series or text**, these networks use their internal state (memory) to process sequences of inputs.



Machado de Assis Bot with RNN - GRU

The *robot writer* model is a **Recurrent Neural network (RNN —GRU)**. To obtain the final AI model, 3.5 million parameters were trained with a **120-letter sequence** from seven of his books: *Memorias Posthumas de Braz Cubas, Dom Casmurro, Quincas Borba, Papeis Avulsos, A Mão e a Luva, Esaú e Jacob*, and *Memorial de Ayres*.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (128, None, 64)           7488
_________________________________________________________________
gru (GRU)                    (128, None, 1026)         3361176
_________________________________________________________________
dense (Dense)                (128, None, 117)          120159
=================================================================
Total params: 3,488,823
Trainable params: 3,488,823
Non-trainable params: 0
_________________________________________________________________
```



A LUVA DE CASMURRO II

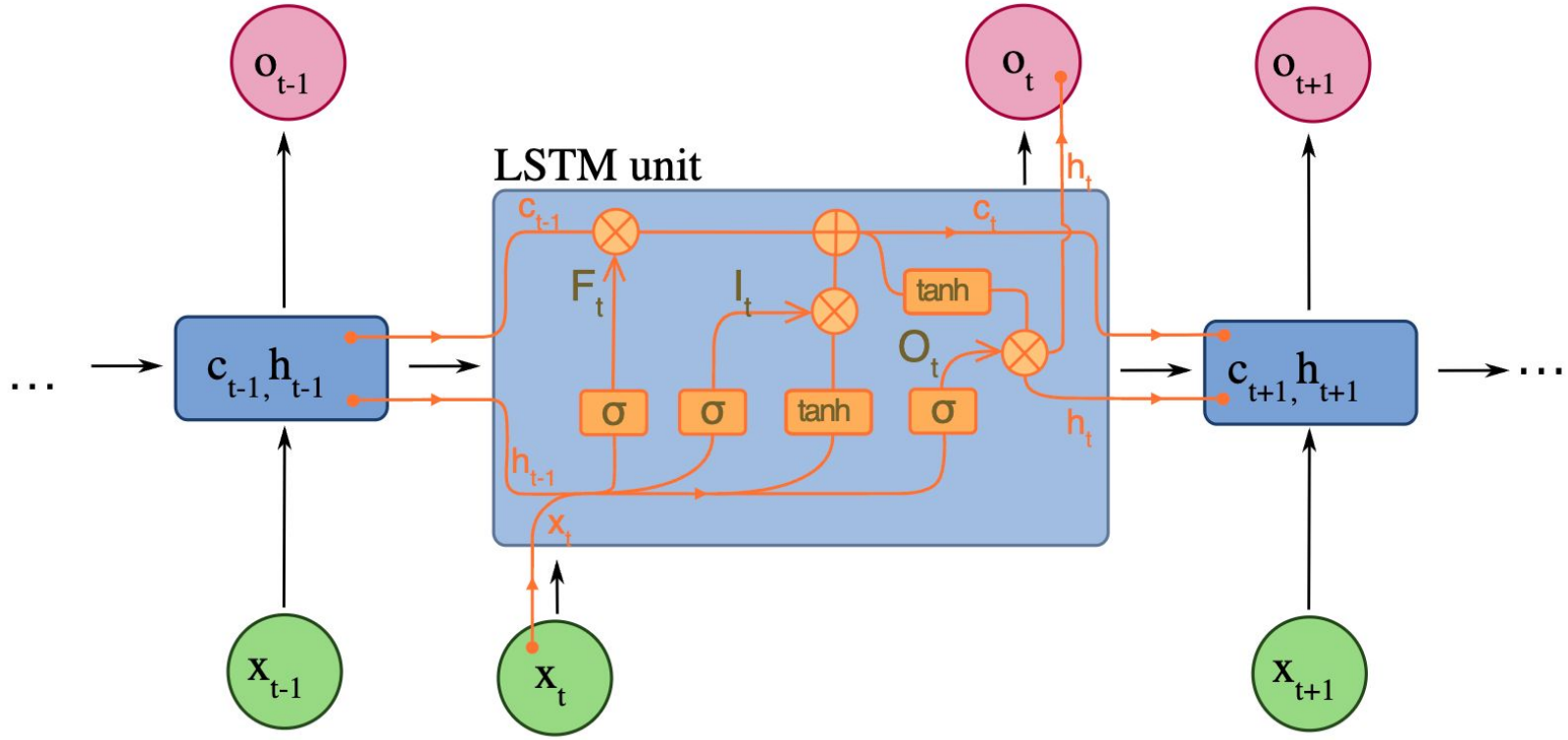
A missa do coupé e um presente e o governo devia cazar logo no papel, a morte do autor, e todos os seus considerados de alegria. Era um espirito de vinte e cinco annos, e eu não estou alguns passos no cerebro, como de outra cousa. Deus me disse:

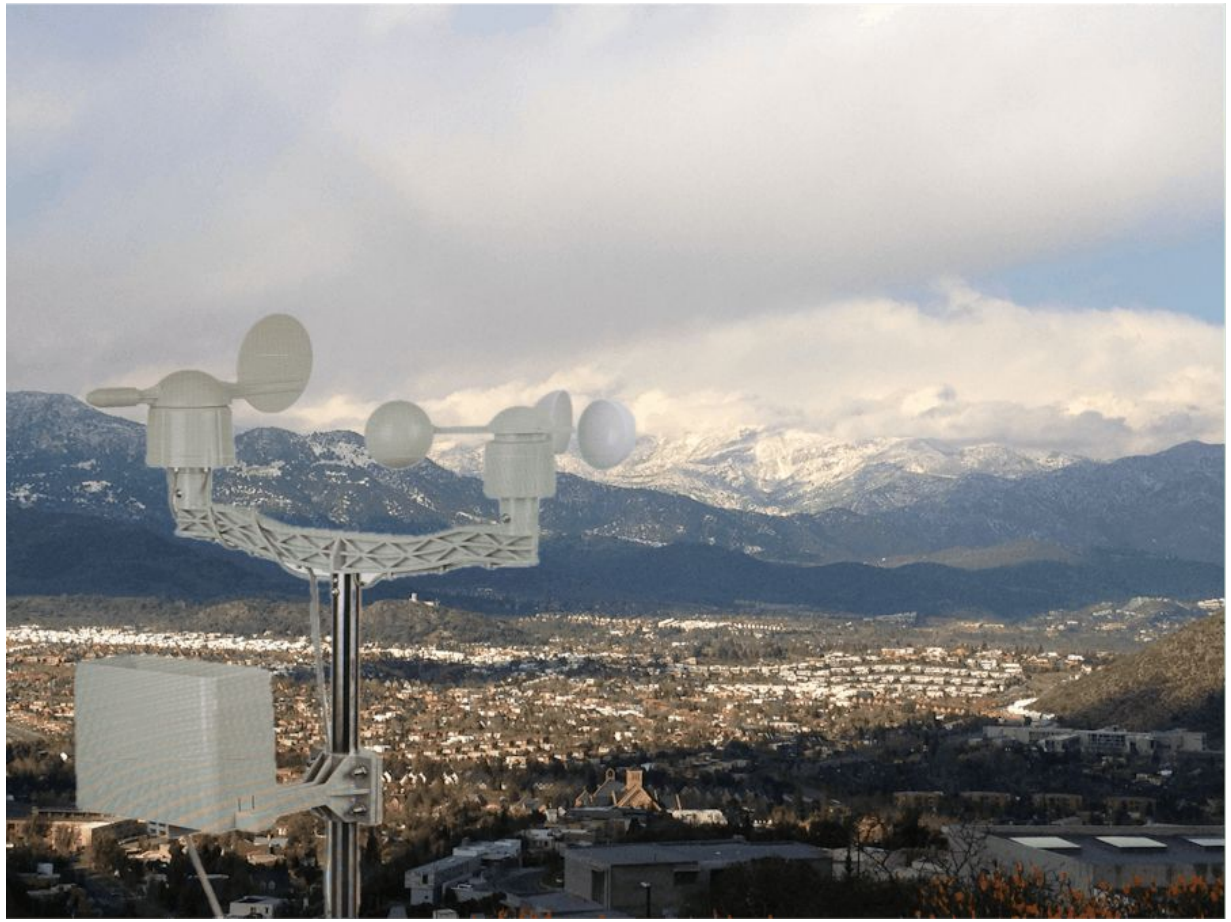
--Não digo que não. Se eu tivesse a intenção de um probosito. Palha acudiu a mulher, não havia nada. A noite vinha tambem para o seminario, tinha o aspecto do partido recto e de restaurar a minha mãe e do pae, pela primeira vez, a menor destinada a dispensar o chapéo, esperou que não vinhas com as suas mãos de creanças. A manhã della chegasse a baroneza e a maneira desta divida. Parece que é casada.

--Está bom, perdoa-lhe de todos os lados, a vida de que o comprar para o meu quarto de hora, e contavam com o fim de a anterior, e, a parede pouco tempo a alma de pessoas que definitivamente lhe interessam a menos para mim. De quando em quando, esses dous annos de conversação para o fim de deixar nenhuma pessoa que se dispersasse; mas não falo de uma cousa nem lhe pedia com a mão tremula, como se ella quizesse. Eu, apertando-lhe a mão, aliás o principio do governo, a proposito disso, com a desattenção de Estevão, e eu começou a aborrecel-o, e a solidão podia ser melhor, e a sympathia colloca da mãe, e não se sabe calar o enterro no meio do lagem, o que iam-se apanhados no chão, e para a mulher, não tendo visto, nem a mesma cousa.

Deep Learning models (or artificial neural networks)

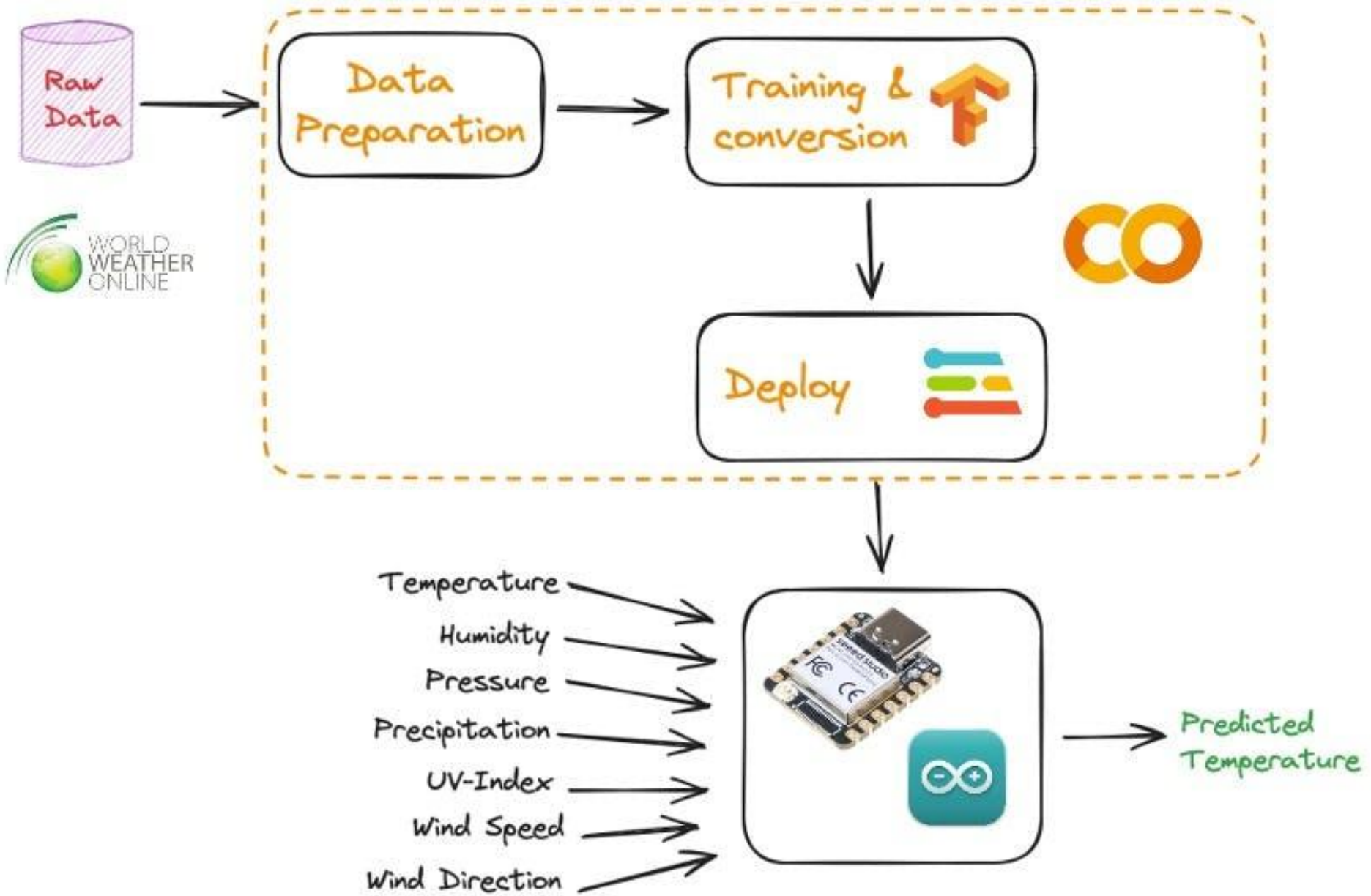
Long Short-Term Memory (LSTM): A type of RNN that can learn over long sequences without losing information, effectively managing long-term dependencies.

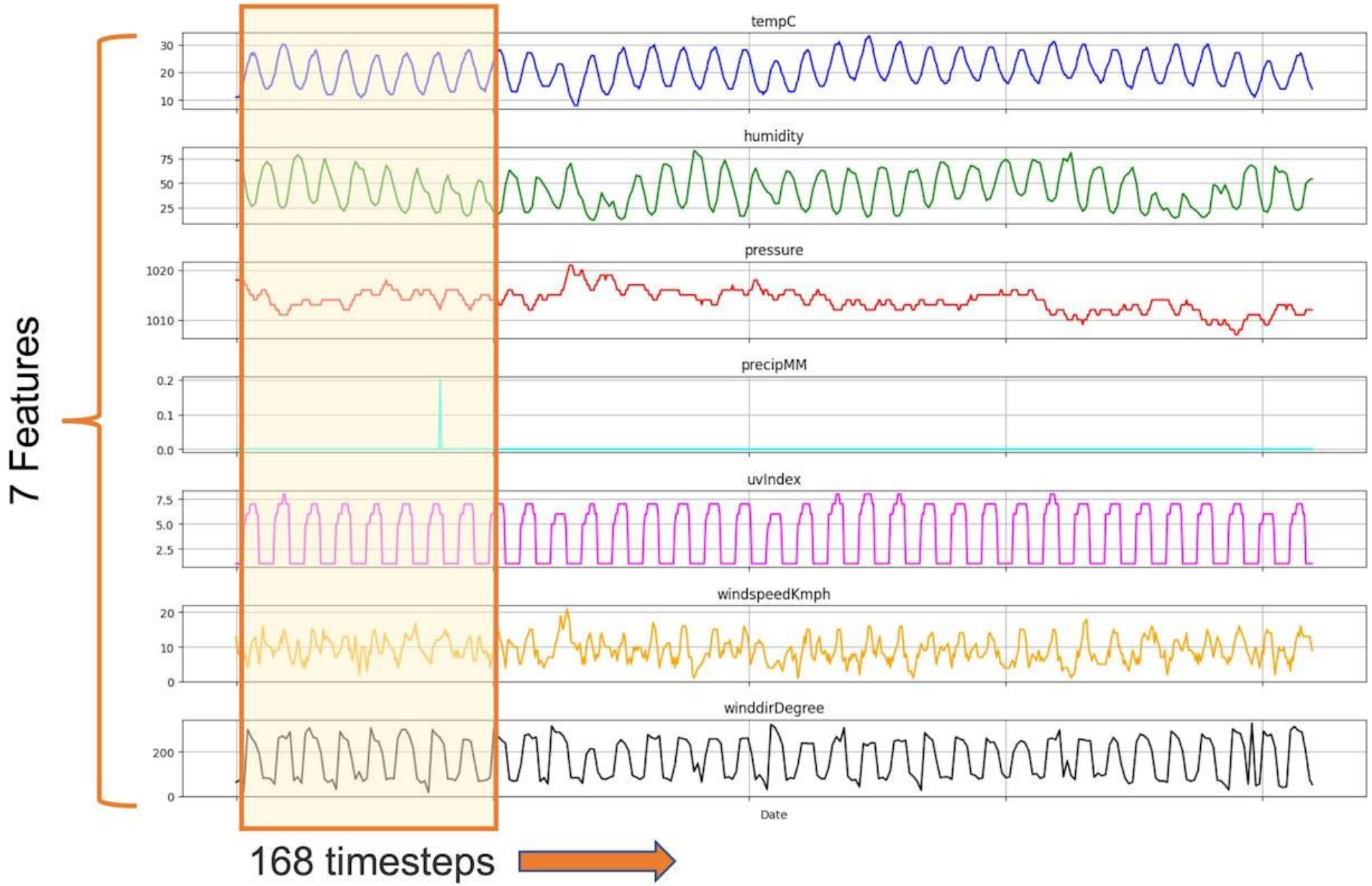


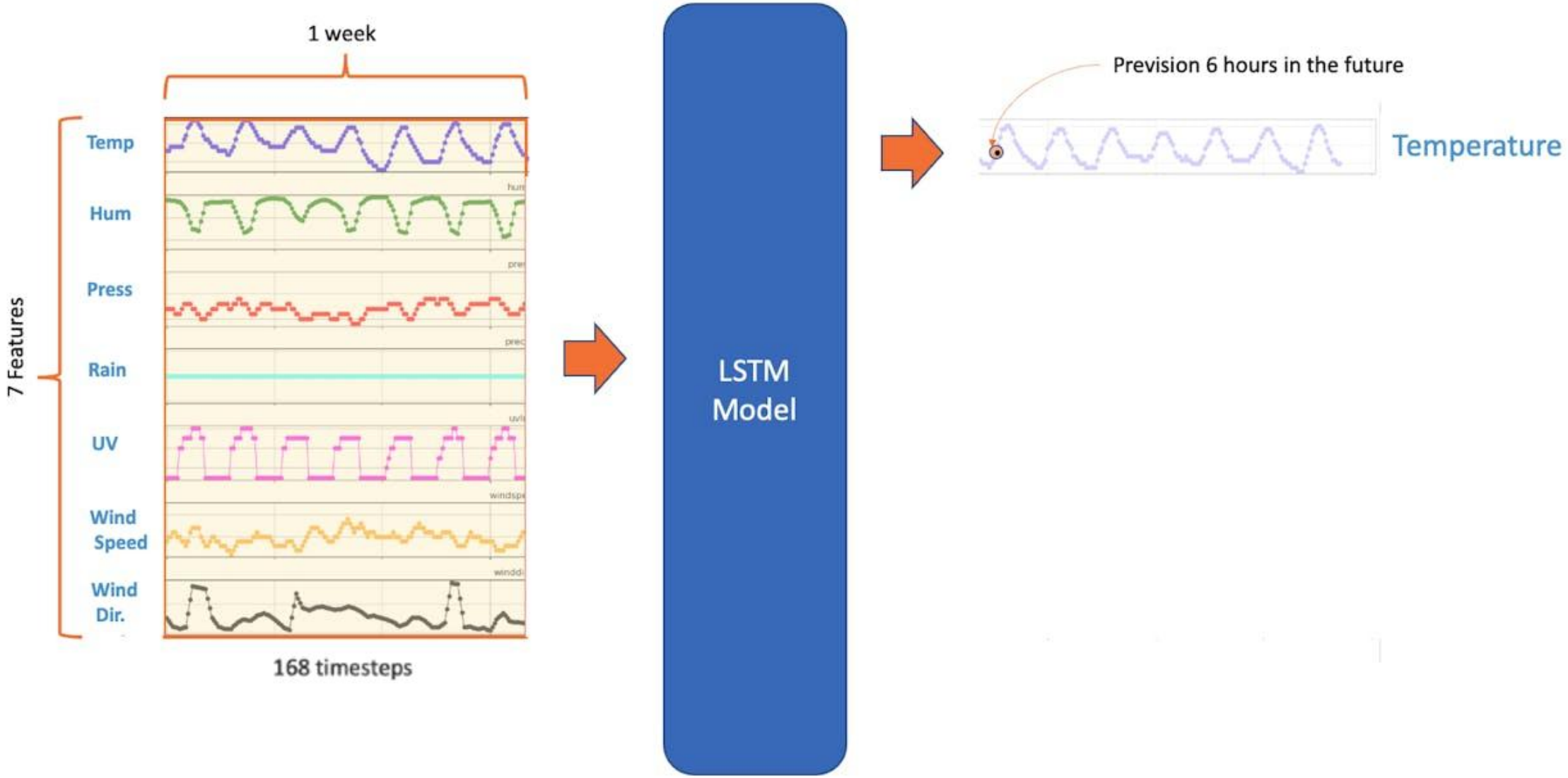


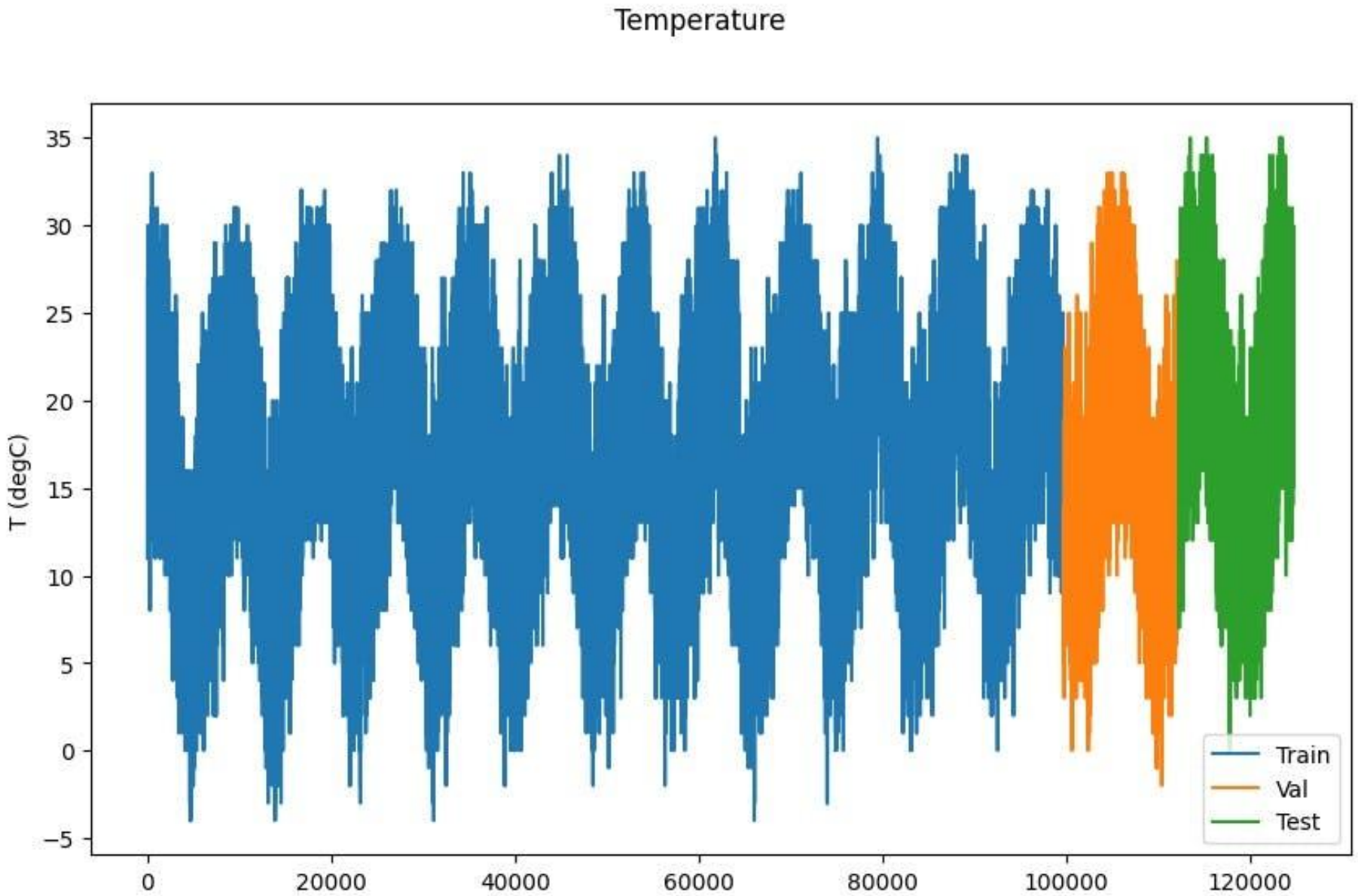
Temperature Prediction using an LSTM model

*The trained LSTM model will be converted with TFLite-Micro and **Edge Impulse Python SDK** and deployed on an XIAO ESP32S3.*


Raw Data
WORLD WEATHER ONLINE
Data Preparation
Training & conversion
Deploy
Temperature
Humidity
Pressure
Precipitation
UV-Index
Wind Speed
Wind Direction
Predicted Temperature


7 Features
tempC
30
20
10
humidity
75
50
25
pressure
1020
1010
precipMM
0.2
0.1
0.0
uvIndex
7.5
5.0
2.5
windspeedKmph
20
10
0
winddirDegree
200
0
Date
168 timesteps

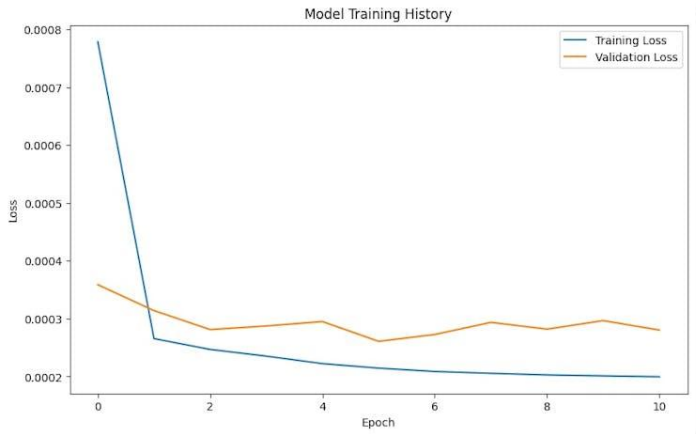

1 week
7 Features
Temp
Hum
Press
Rain
UV
Wind Speed
Wind Dir.
168 timesteps
LSTM Model
Prevision 6 hours in the future
Temperature


Temperature
T (degC)
35
30
25
20
15
10
5
0
−5
0
20000
40000
60000
80000
100000
120000
Train
Val
Test

```python
model = Sequential([
    LSTM(128,
         input_shape=(n_steps, X_train.shape[2])),
    Dense(1)
])
```

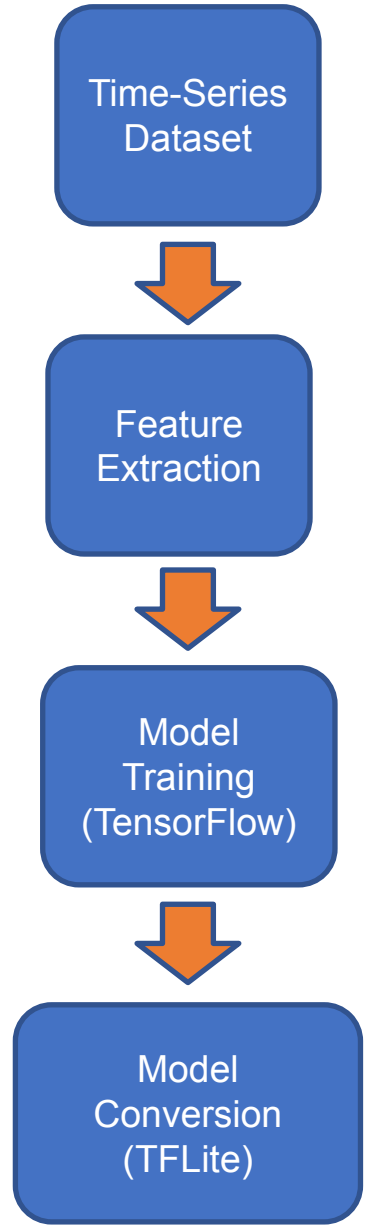
```python
model.compile(optimizer='adam', loss='mse')
```

```python
history = model.fit(
    X_train, y_train,
    validation_data=(X_val, y_val),
    epochs=20,
    batch_size=32,
    callbacks=[early_stopping]
)
```

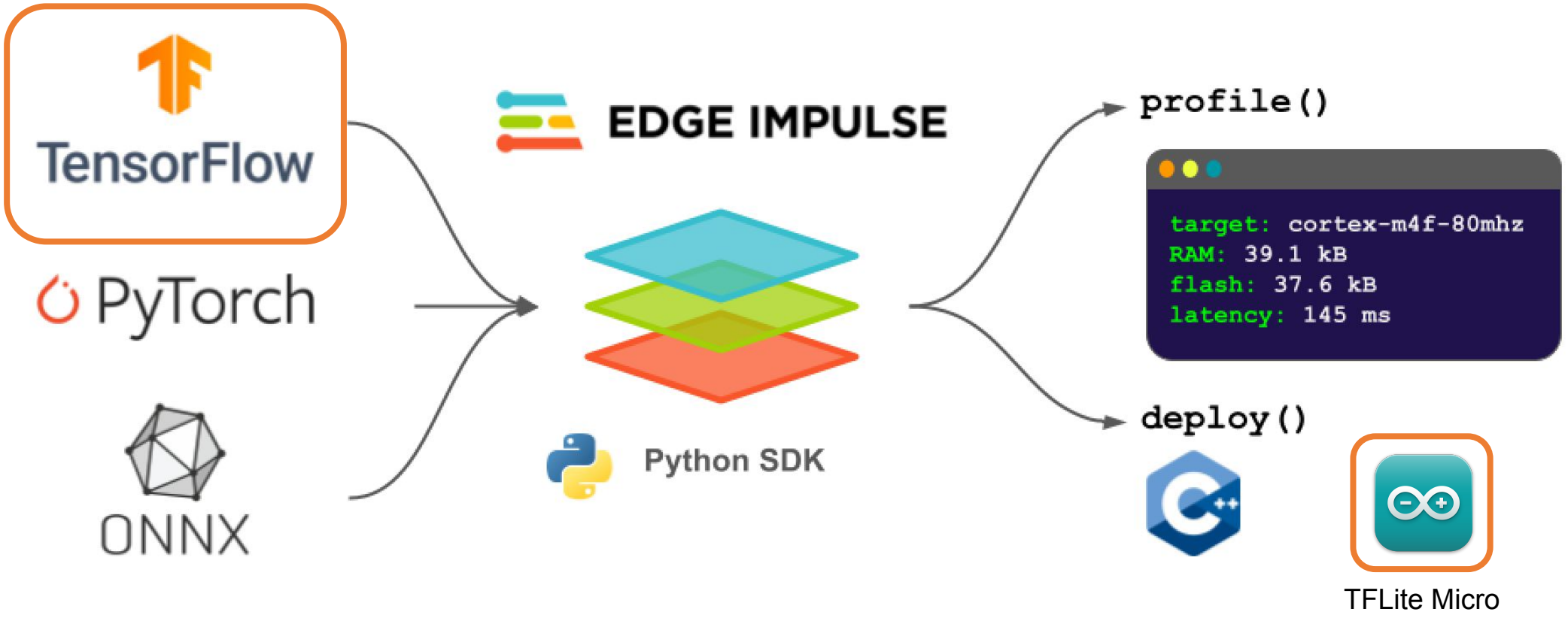



```python
prediccion = model.predict(X_test)
```

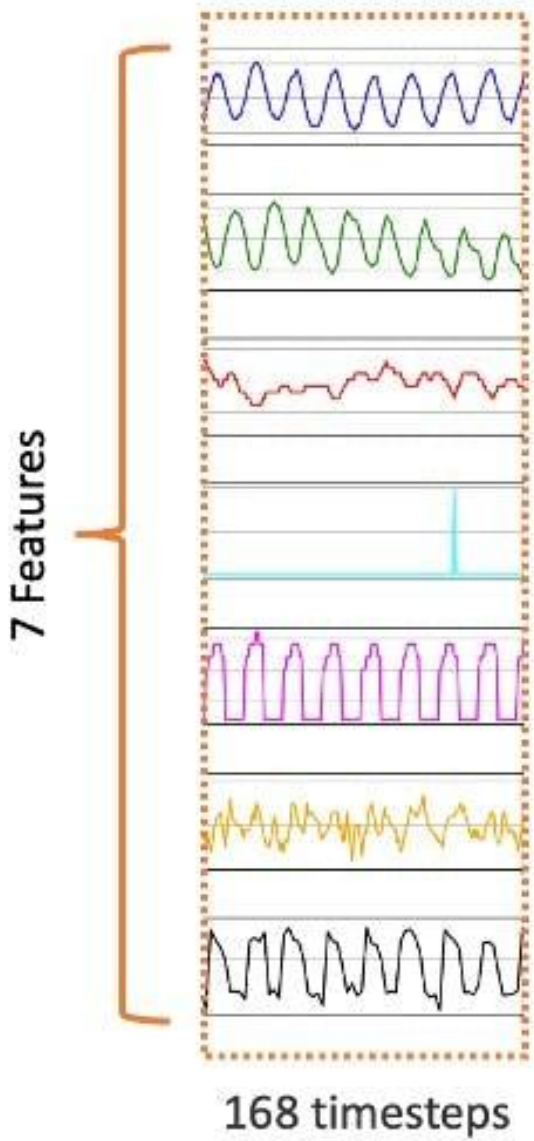
```python
converter = tf.lite.TFLiteConverter.from_saved_model(MODEL_DIR)
tflite_model = converter.convert()

# Save the converted model to file
tflite_model_file = 'converted_model.tflite'
with open(tflite_model_file, 'wb') as f:
    f.write(tflite_model)
```



Edge Impulse Python SDK



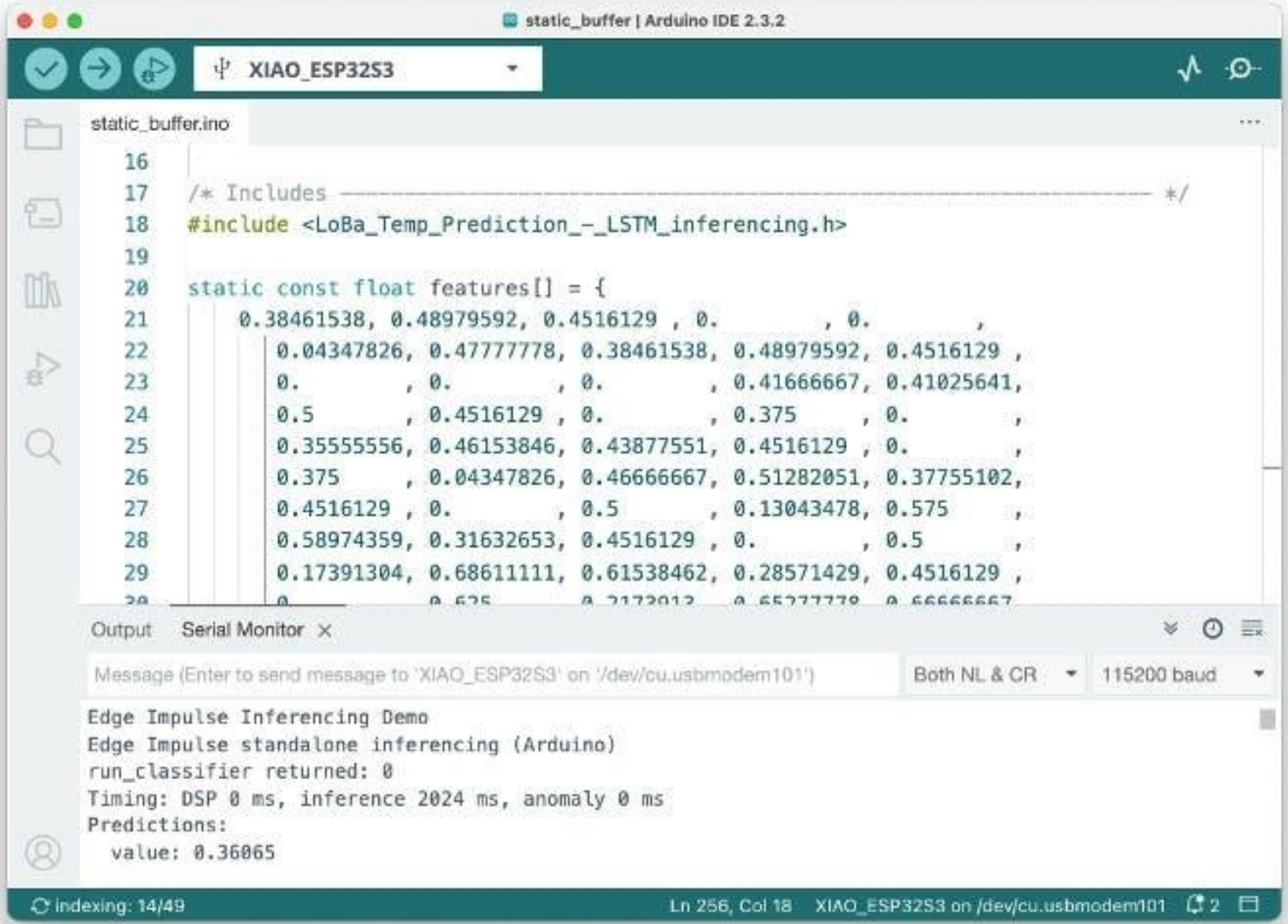
converted_model.tflite → EI Python SDK → lstm_float32_model.zip

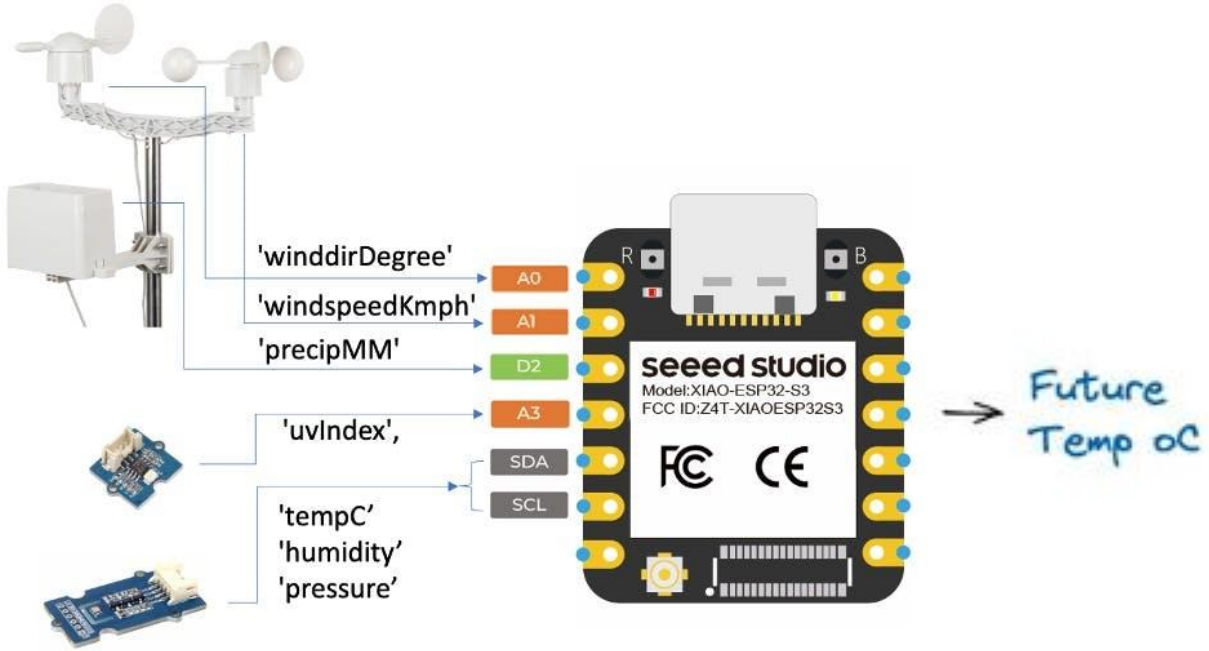


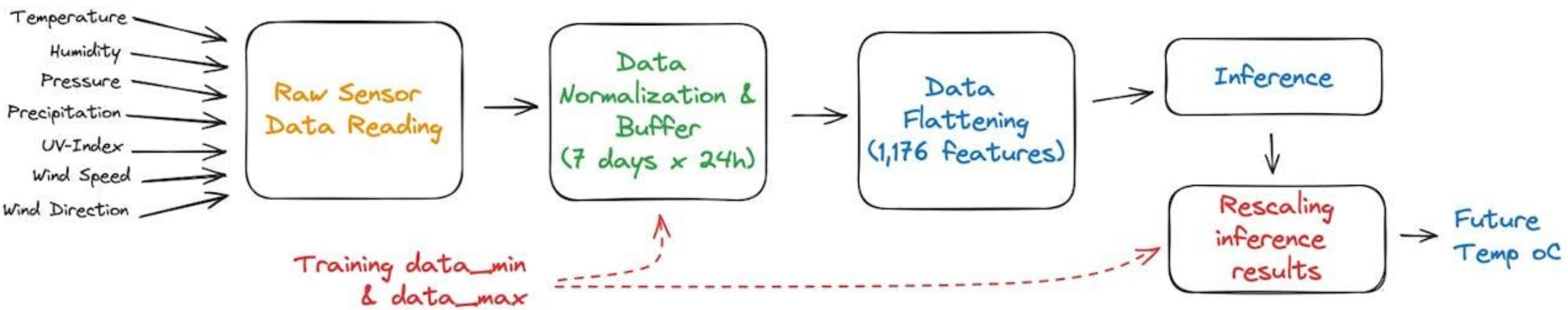

```
[0.38461538, 0.48979592, 0.4516129 , 0.        , 0.        ,
 0.04347826, 0.47777778, 0.38461538, 0.48979592, 0.4516129 ,
 0.        , 0.        , 0.        , 0.41666667, 0.41025641,
 0.5       , 0.4516129 , 0.        , 0.375     , 0.        ,
 0.35555556, 0.46153846, 0.43877551, 0.4516129 , 0.        ,
 0.375     , 0.04347826, 0.46666667, 0.51282051, 0.37755102,
 0.4516129 , 0.        , 0.5       , 0.13043478, 0.575     ,
 0.58974359, 0.31632653, 0.4516129 , 0.        , 0.5       ,
 0.17391304, 0.68611111, 0.61538462, 0.28571429, 0.4516129 ,
 0.        , 0.625     , 0.2173913 , 0.65277778, 0.66666667,
 0.26530612, 0.4516129 , 0.        , 0.625     , 0.30434783,
 0.61666667, 0.71794872, 0.24489796, 0.41935484, 0.        ,
 0.625     , 0.34782609, 0.58333333, 0.74358974, 0.2244898 ,
 0.41935484, 0.        , 0.625     , 0.47826087, 0.57777778,
 0.76923077, 0.21428571, 0.41935484, 0.        , 0.75      ,
 0.60869565, 0.575     , 0.74358974, 0.20408163, 0.38709677,
 0.        , 0.75      , 0.73913043, 0.57222222, 0.69230769,
 0.2244898 , 0.41935484, 0.        , 0.625     , 0.69565217,
```

1,176 Features

```
Edge Impulse Inferencing Demo
Edge Impulse standalone inferencing (Arduino)
run_classifier returned: 0
Timing: DSP 0 ms, inference 2024 ms, anomaly 0 ms
Predictions:
  value: 0.36065
```


static_buffer | Arduino IDE 2.3.2
XIAO_ESP32S3
static_buffer.ino
16
17 /* Includes ---------- */
18 #include <LoBa_Temp_Prediction_-_LSTM_inferencing.h>
19
20 static const float features[] = {
21 0.38461538, 0.48979592, 0.4516129 , 0. , 0. ,
22 0.04347826, 0.47777778, 0.38461538, 0.48979592, 0.4516129 ,
23 0. , 0. , 0. , 0.41666667, 0.41025641,
24 0.5 , 0.4516129 , 0. , 0.375 , 0. ,
25 0.35555556, 0.46153846, 0.43877551, 0.4516129 , 0. ,
26 0.375 , 0.04347826, 0.46666667, 0.51282051, 0.37755102,
27 0.4516129 , 0. , 0.5 , 0.13043478, 0.575 ,
28 0.58974359, 0.31632653, 0.4516129 , 0. , 0.5 ,
29 0.17391304, 0.68611111, 0.61538462, 0.28571429, 0.4516129 ,
Output
Serial Monitor
Message (Enter to send message to 'XIAO_ESP32S3' on '/dev/cu.usbmodem101')
Both NL & CR
115200 baud
Edge Impulse Inferencing Demo
Edge Impulse standalone inferencing (Arduino)
run_classifier returned: 0
Timing: DSP 0 ms, inference 2024 ms, anomaly 0 ms
Predictions:
value: 0.36065
indexing: 14/49
Ln 256, Col 18
XIAO_ESP32S3 on /dev/cu.usbmodem101
2


'winddirDegree'
'windspeedKmph'
'precipMM'
'uvIndex',
'tempC'
'humidity'
'pressure'
A0
A1
D2
A3
SDA
SCL
R
B
seeed studio
Model:XIAO-ESP32-S3
FCC ID:Z4T-XIAOESP32S3
Future
Temp oC


Temperature
Humidity
Pressure
Precipitation
UV-Index
Wind Speed
Wind Direction
Raw Sensor Data Reading
Data Normalization & Buffer (7 days x 24h)
Data Flattening (1,176 features)
Inference
Rescaling inference results
Future Temp oC
Training data_min & data_max

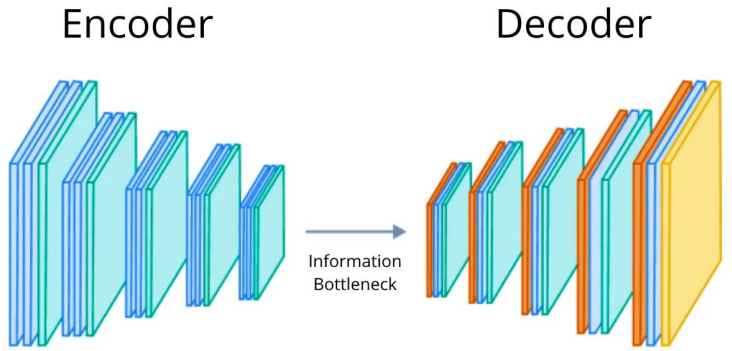
LLM / SLM

Large Language Model / Small Language Models

LLMs are **specialized deep learning models designed to understand and generate human language**, used for tasks like translation, summarization, and generating human-like text responses. SLMs are the same, but use a simpler, less resource-intensive approach (smaller in size).

Deep Learning models (or artificial neural networks)

- **Autoencoders**: Used primarily for unsupervised learning tasks such as dimensionality reduction and feature extraction, autoencoders learn to compress data from the input layer into a shorter code and then reconstruct the output from this representation.



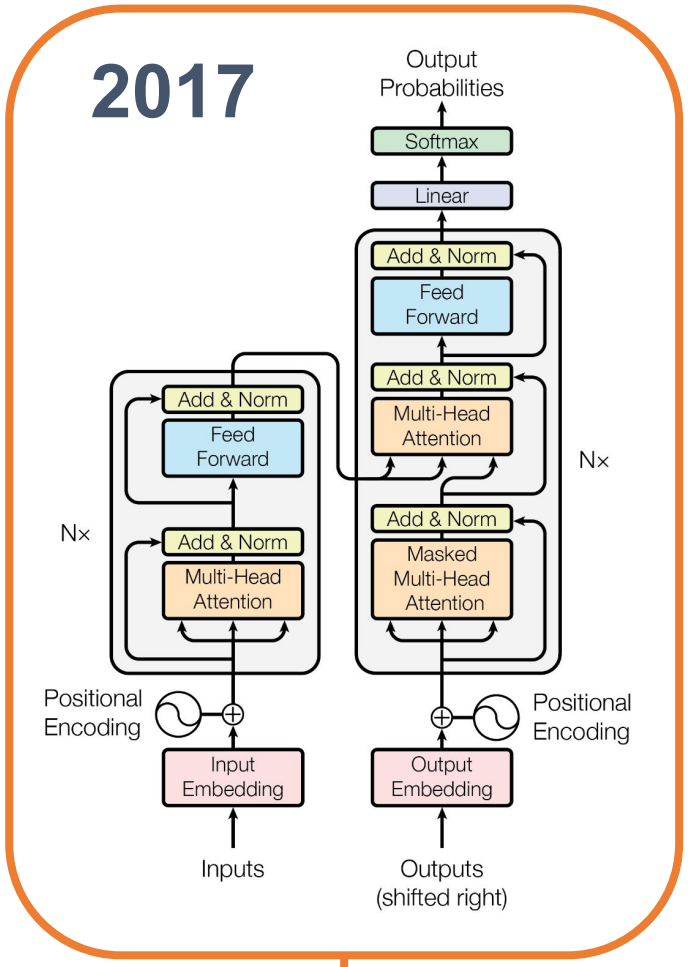
- **Transformer Models**: Highly effective in handling sequences, transformers use mechanisms like self-attention to weigh the importance of different words in a sentence, regardless of their position. The Transformer architecture, while innovative, can be seen as a derivative of earlier deep learning models, particularly those based on the concept of sequence modeling. However, the most direct lineage can be traced to the sequence-to-sequence (seq2seq) models that utilize **encoder-decoder** architectures. These earlier seq2seq models were often built using **recurrent neural networks (RNNs)** or their more advanced variants like **LSTMs (Long Short-Term Memory Networks) or GRUs (Gated Recurrent Units).**

LLM/SLM – Large /Small Language Model

Large Language Models (LLMs) and SLMs are advanced neural networks based on the Transformer architecture that excel in understanding and generating human language. They represent a significant evolution from earlier sequence-based models like LSTMs, which surpass them in handling long-range dependencies and parallel processing efficiency.

Prof. Jesus's Presentation about IA: 

Transformers to LLMs and SLMs

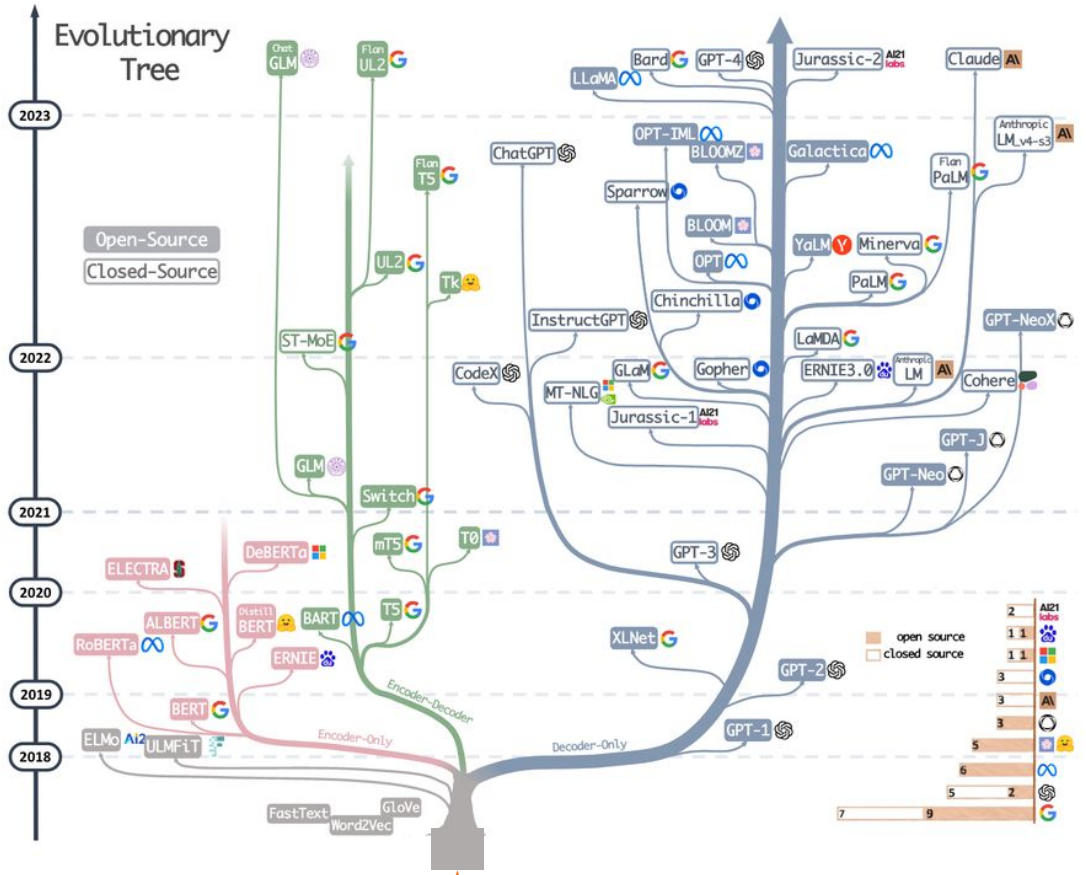


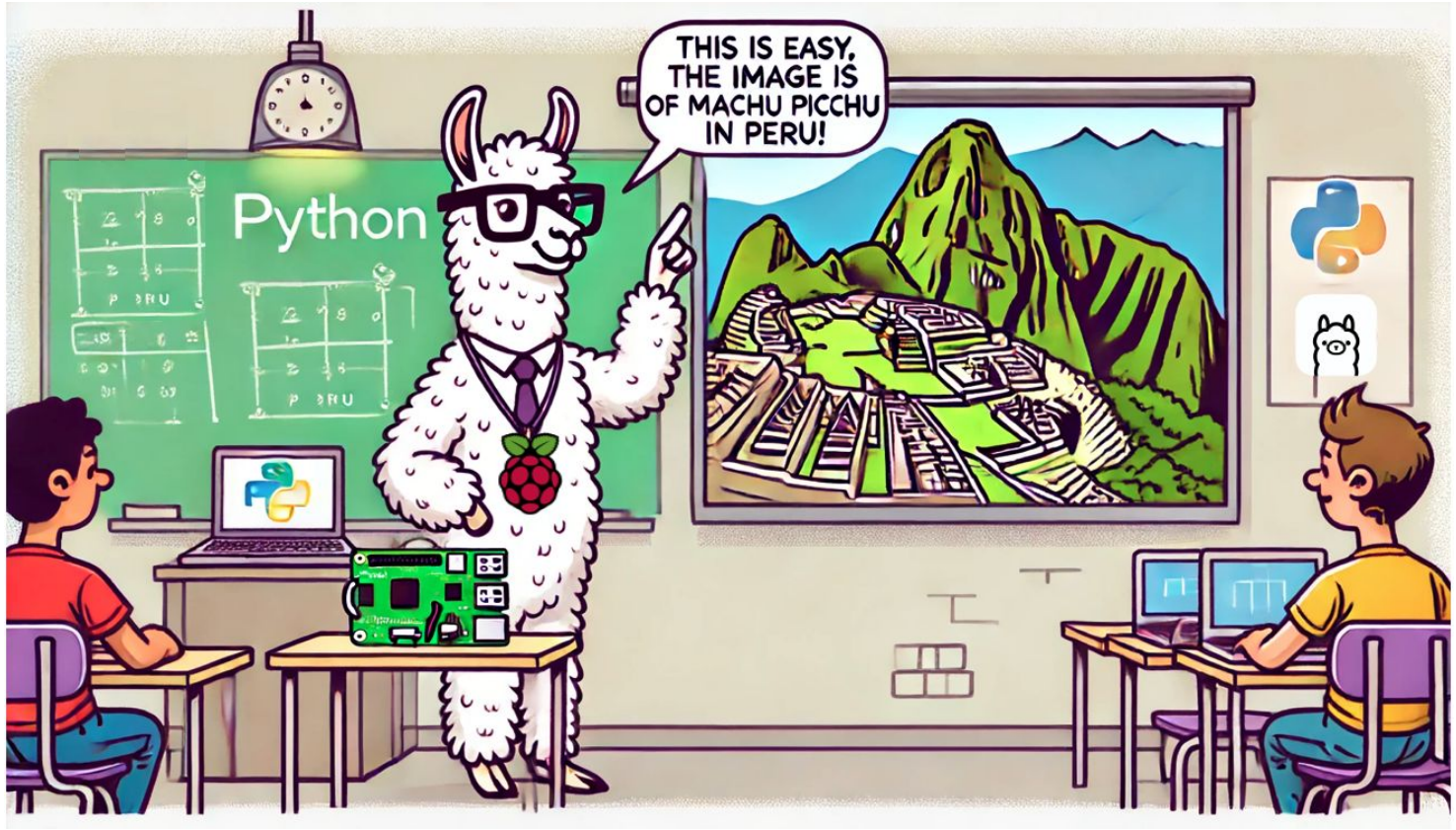
Open

Phi | Mixtral | LlaMa

Closed

Gemini | GPT | Claude

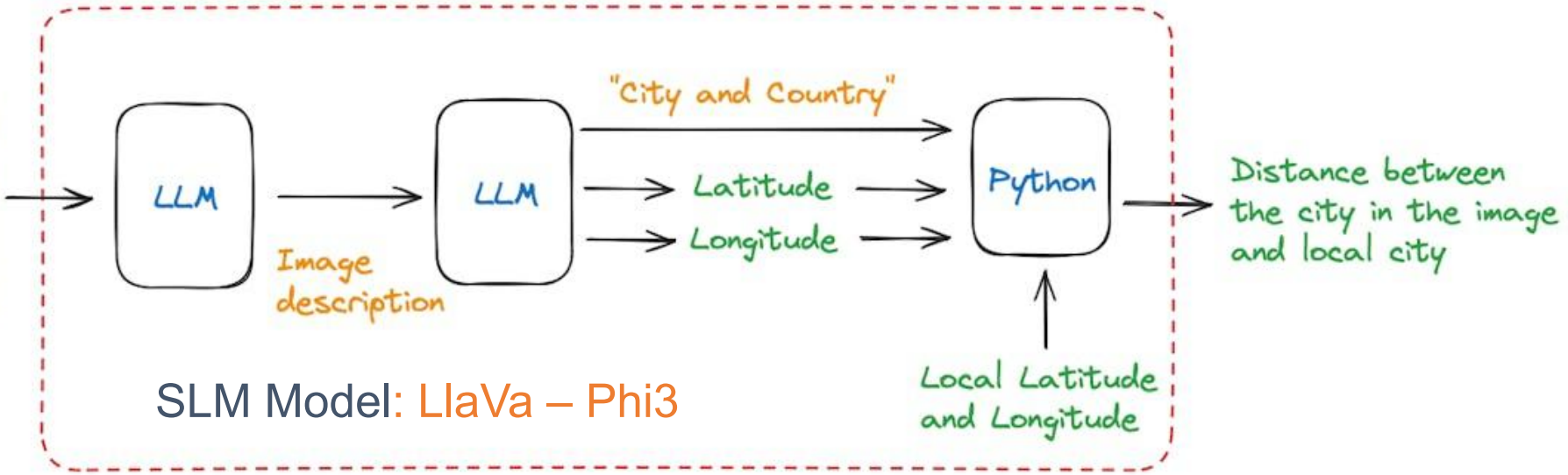




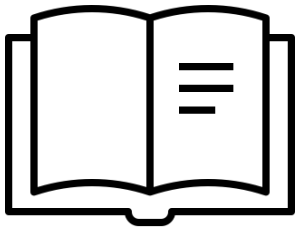
Running Large Language Models on Raspberry Pi at the Edge

Transform a Raspberry Pi into a powerful AI hub, running SLMs for real-time, on-site data analysis and insights using Ollama and Python.

The image shows Machu Picchu with lat: -13.16 and long: -72.54, located in Peru and about 2,262 kilometers away from Santiago, Chile.



llava-phi-3 is a LLaVA model (**L**arge **La**nguage and **V**ision **A**ssistant) fine-tuned from Microsoft Phi-3 mini



~ 350 pages

~ 300 words/page

1 word = ~ 1.4 token

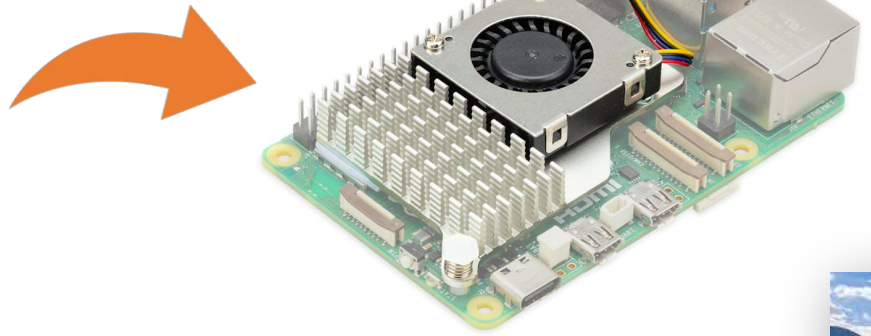
A **4-bit** quantized 3.8 billion parameter * language model trained on 3.3 trillion tokens**, whose overall performance, as measured by both academic benchmarks and internal testing, rivals that of models such as Mixtral 8x7B and GPT-3.5

* 2.4 GB

** 22.5 Million books - 17% of all books written in the world

llava-phi-3 (2.9 GB)

Ollama



```
mjrovai@rpi-5: ~/Documents/OLLAMA
Help
ute.

/Documents/OLLAMA $
/Documents/OLLAMA $ python calc_distance_image.py /
home/mjrovai/Documents/OLLAMA/image_test_1.jpg

 The image shows Paris, with lat:48.86 and long: 2.35, located in
France and about 11,630 kilometers away from Santiago, Chile.

 [INFO] ==> The code (running llava-phi3), took 232.60845186299412
 seconds to execute.

mjrovai@rpi-5:~/Documents/OLLAMA $
```

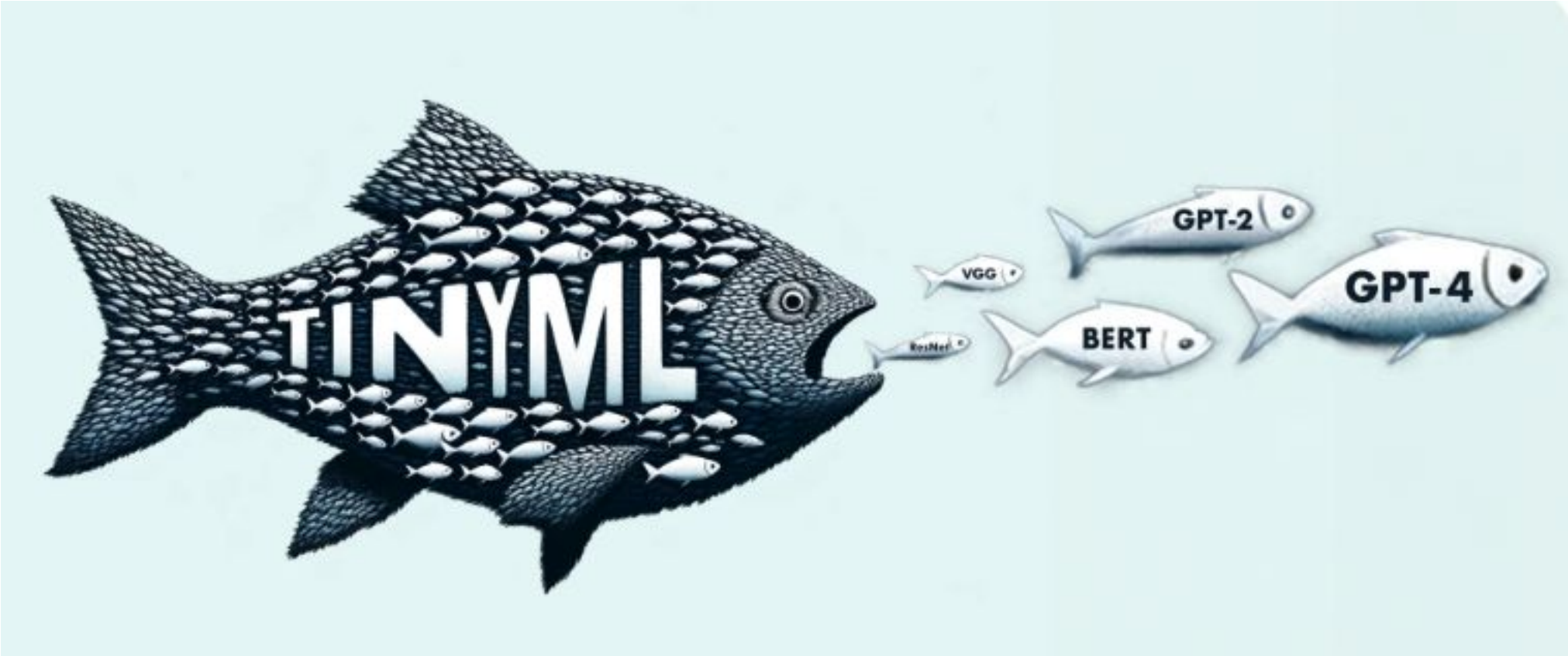
```
mjrovai@rpi-5: ~
File  Edit  Tabs  Help
>>> Awswer with one short sentence, what is the capital of France and its distanc
... e in Km from Santiago, Chile
 The capital of France is Paris and it is around 12,674 kilometers away
from Santiago, Chile.

total duration:       13.860074968s
load duration:        1.537039ms
prompt eval count:    27 token(s)
prompt eval duration: 5.925386s
prompt eval rate:     4.56 tokens/s
eval count:           26 token(s)
eval duration:        7.539223s
eval rate:            3.45 tokens/s
>>> Send a message (/? for help)
```

(13 seconds)

```
mjrovai@rpi-5: ~/Documents/OLLAMA
Help
/Documents/OLLAMA $
/Documents/OLLAMA $ python calc_distance_image.py
/home/mjrovai/Documents/OLLAMA/image_test_3.jpg

 The image shows Machu Picchu, with lat:-13.16 and long: -72.54,
 located in Peru and about 2,250 kilometers away from Santiago,
Chile.

 [INFO] ==> The code (running llava-phi3), took 267.579568572007
7 seconds to execute.

mjrovai@rpi-5:~/Documents/OLLAMA $
```

(4 minutes)

TinyML: Why the Future of Machine Learning is Tiny and Bright



Shvetank Prakash, Emil Njor, Colby Banbury, Matthew Stewart, Vijay Janapa Reddi

To learn more …

Online Courses

Harvard School of Engineering and Applied Sciences - CS249r: Tiny Machine Learning
Professional Certificate in Tiny Machine Learning (TinyML) – edX/Harvard
Introduction to Embedded Machine Learning - Coursera/Edge Impulse
Computer Vision with Embedded Machine Learning - Coursera/Edge Impulse
UNIFEI-IESTI01 TinyML: "Machine Learning for Embedding Devices"

Books

"Python for Data Analysis" by Wes McKinney
"Deep Learning with Python" by François Chollet - GitHub Notebooks
"TinyML" by Pete Warden and Daniel Situnayake
"TinyML Cookbook 2nd Edition" by Gian Marco Iodice
"Technical Strategy for AI Engineers, In the Era of Deep Learning" by Andrew Ng
"AI at the Edge" book by Daniel Situnayake and Jenny Plunkett
"XIAO: Big Power, Small Board" by Lei Feng and Marcelo Rovai
"MACHINE LEARNING SYSTEMS for TinyML" by a collaborative effort

Projects Repository

Edge Impulse Expert Network

On the TinyML4D website, You can find lots of educational materials on TinyML. They are all free and open-source for educational uses – we ask that if you use the material, please cite them! TinyML4D is an initiative to make TinyML education available to everyone globally.

Thanks

UNIFEI