

AIOT

Nhận diện đồ vật bằng bounding box

Tên: Lê Âu Xuân Dương
MSSV: 19120488



code arduino deploy :

<https://drive.google.com/drive/folders/15U9S6c9FL8AjbJWOaemhH8xYdTvxJoDU?usp=sharing>

Giảng viên: Nguyễn Đức
Hoàng Hạ Tài liệu tham khảo: Edge
impulse -

<https://docs.edgeimpulse.com/docs/tutorials/object-detection>

BÀI TOÁN 01

DATA 02

TABLE OF CONTENTS


04 MODEL

05 DEPLOY

01

BÀI TOÁN





Phát hiện vật thể bằng bounding box

*Là một loại bài toán phổ biến ngày nay
Cần nắm kiến thức về: image
classification, kiến trúc convolutional
neural network và CNN*





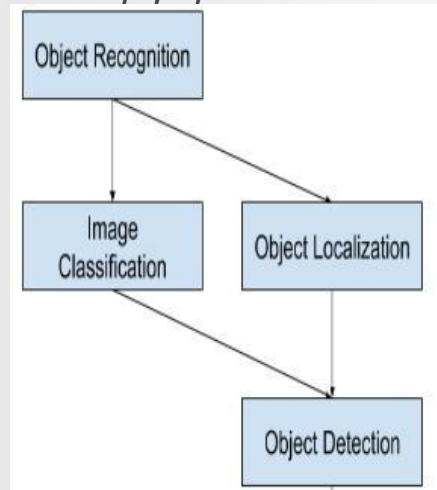
Phát hiện vật thể bằng bounding box

*image
classification
object
localization
object detection*



Phát hiện vật thể bằng bounding box

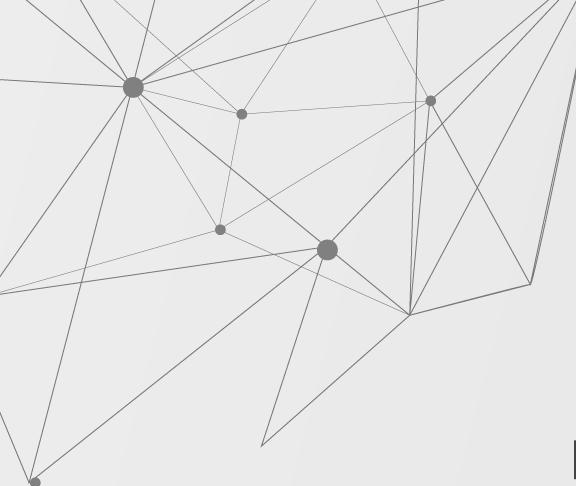
object detection: vẽ bounding box, gán





02

DATA



Công cụ hỗ trợ

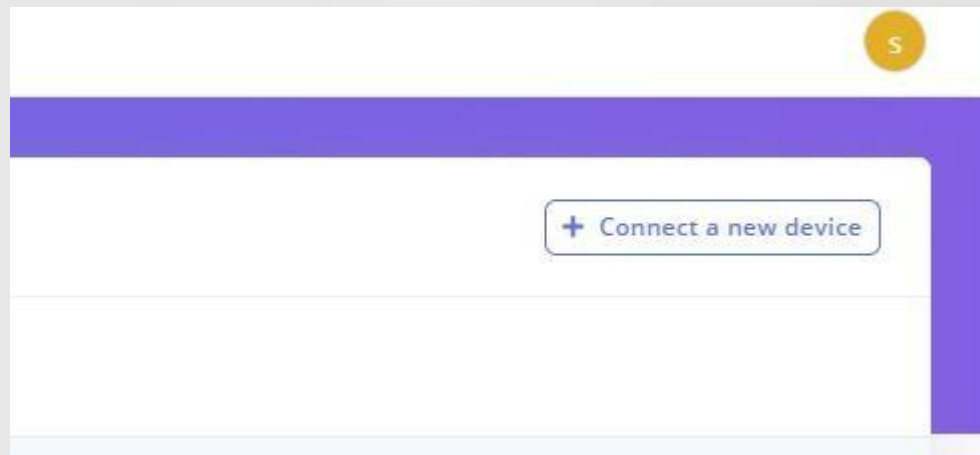
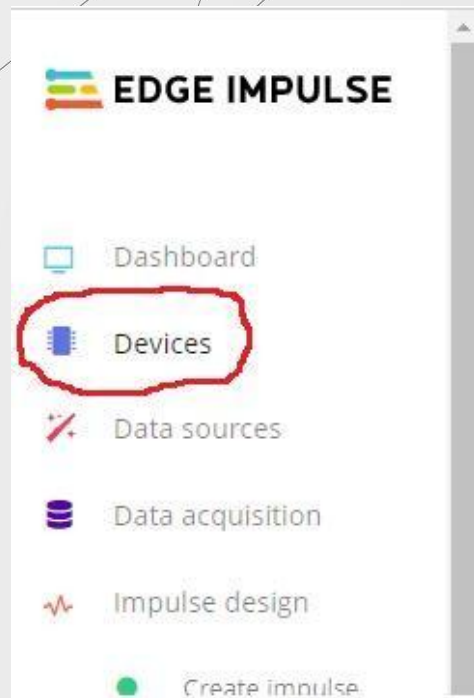
Edge impluse

Hỗ trợ
developers Xây
dựng, triển khai
mở rộng quy mô
cho machine
learning.
Hỗ trợ xây dựng
trên
thiết bị nhúng



Thu thập dữ liệu

Kết nối thiết bị



DATA

85%

Training

15%

Test

DATA COLLECTED

62 items



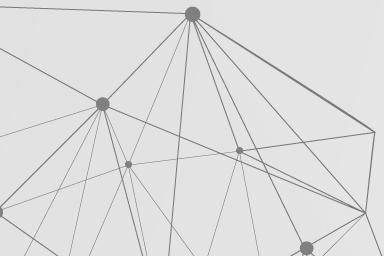
TRAIN / TEST SPLIT

85% / 15% ?



DATA

Labeling queue





03

Model

Create impulse

Image data

Input axes

image

Image width

320

Image height

320

Resize mode

Fit shortest axis

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 (chén, ly)

Output features

2 (chén, ly)

Save Impulse

MODEL

Create impulse

Image data

Input axes

image

Image width

48

Image height

48

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 (chén, ly)

Output features

2 (chén, ly)

Save Impulse

MODEL

Trực quan hóa tập
dữ liệu



MODEL

Training setting

Number of training cycles ⓘ

25

Learning rate ⓘ

0.015

Validation set size ⓘ

20

%

Neural network architecture

Input layer (307,200 features)



MobileNetV2 SSD FPN-Lite 320x320

Choose a different model

Output layer (2 classes)

Training output

Model

Model version: ⓘ

Unoptimized (float32) ▼

Last training performance (validation set)



PRECISION SCORE

92.4%

On-device performance ⓘ



INFERRING TIME

345 ms.



FLASH USAGE

11.0M

Training setting

Neural Network settings

Training settings

Number of training cycles ⓘ

60

Learning rate ⓘ

0.001

Validation set size ⓘ

20


%

Data augmentation ⓘ

☒

Neural network architecture

Input layer (6.912 features)


FOMO (Faster Objects, More Objects) MobileNetV2 0.1

Choose a different model

Output layer (2 classes)

Start training


MODEL

Model

Model version: ⓘ

Quantized (int8) ▾

Last training performance (validation set)


 F1 SCORE

87.0%


Confusion matrix (validation set)

	BACKGROUND	CHÉN	LY
BACKGROUND	99.3%	0.4%	0.2%
CHÉN	27.3%	72.7%	0%
LY	0%	0%	100%
F1 SCORE	0.99	0.76	0.96


On-device performance ⓘ

 INFERRING TIME

-

 PEAK RAM USAGE

76.0K

 FLASH USAGE

60.8K



MODEL

Mô hình FOMO

Mạng nơ ron TinyML được phát triển bởi Edge impulse. Đem lại hiệu quả khá tốt



Nguồn tham khảo:
tinmoiz.com

04

DEPLOY



Thiết bị sử dụng

Thiết bị sử dụng:
ESP32 CAM



Hỗ trợ xây dựng thư viện, deploy trên nhiều thiết bị

Create library

Turn your impulse into optimized source code that you can run on any device.



C++ library



Arduino library



Cube.MX CMSIS-PAK



WebAssembly



TensorRT library



Ethos-U library



DRP-AI Library

brainchip
MetaTF Model

Meta TF Model



TIDL-RT Library

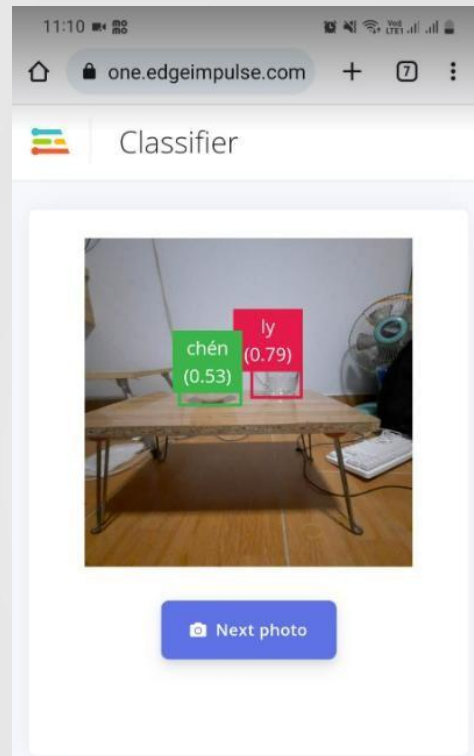
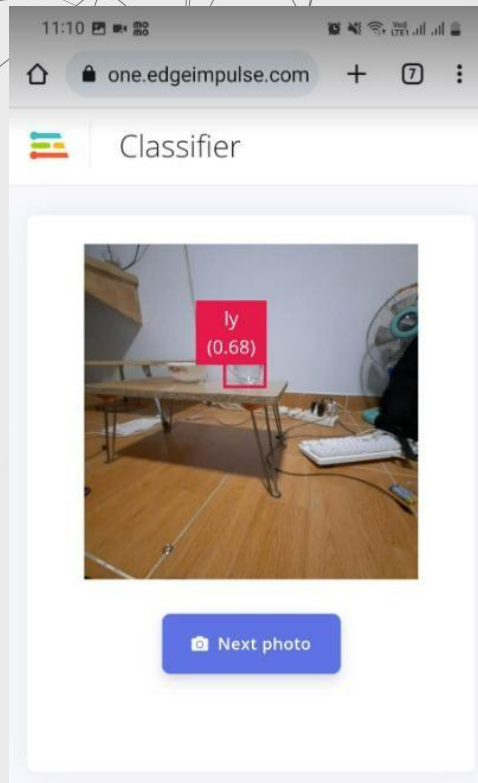


Simplicity Studio Component

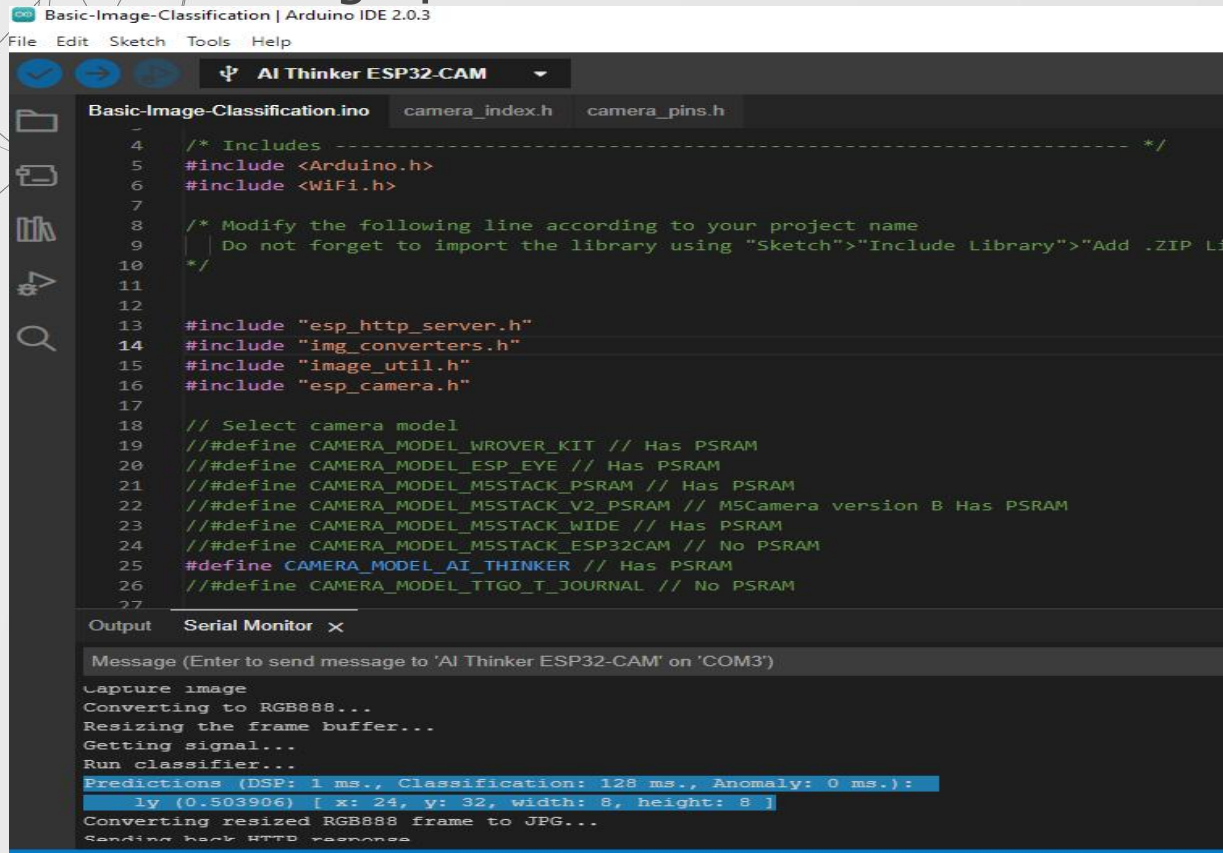


OpenMV library

Thử nghiệm trên thiết bị di động



Thử nghiệm trên ESP32-CAM



Basic-Image-Classification | Arduino IDE 2.0.3

File Edit Sketch Tools Help

AI Thinker ESP32-CAM

Basic-Image-Classification.ino camera_index.h camera_pins.h

```
1
2
3
4  /* Includes ----- */
5  #include <Arduino.h>
6  #include <WiFi.h>
7
8  /* Modify the following line according to your project name
9   | Do not forget to import the library using "Sketch">"Include Library">"Add .ZIP Li
10 */
11
12
13 #include "esp_http_server.h"
14 #include "img_converters.h"
15 #include "image_util.h"
16 #include "esp_camera.h"
17
18 // Select camera model
19 //#define CAMERA_MODEL_WROVER_KIT // Has PSRAM
20 //#define CAMERA_MODEL_ESP_EYE // Has PSRAM
21 //#define CAMERA_MODEL_MSSTACK_PSRAM // Has PSRAM
22 //#define CAMERA_MODEL_MSSTACK_V2_PSRAM // M5Camera version B Has PSRAM
23 //#define CAMERA_MODEL_MSSTACK_WIDE // Has PSRAM
24 //#define CAMERA_MODEL_MSSTACK_ESP32CAM // No PSRAM
25 #define CAMERA_MODEL_AI_THINKER // Has PSRAM
26 //#define CAMERA_MODEL_TTGO_T_JOURNAL // No PSRAM
27
```

Output Serial Monitor x

Message (Enter to send message to 'AI Thinker ESP32-CAM' on 'COM3')

Capture image
Converting to RGB888...
Resizing the frame buffer...
Getting signal...
Run classifier...
Predictions (DSP: 1 ms., Classification: 128 ms., Anomaly: 0 ms.):
ly (0.503906) [x: 24, y: 32, width: 8, height: 8]
Converting resized RGB888 frame to JPG...
Sending back HTTP response



Tài liệu tham khảo:

[1] Cách deploy và sử dụng ESP32-CAM:

<https://github.com/edgeimpulse/example-esp32-cam>

[2] Sửa lỗi: Edge impulse forum

[3] FOMO: <https://docs.edgeimpulse.com/docs/edge-impulse-studio/learning-blocks/object-detection/fomo-object-detection-for-constrained-devices>



Thanks for listening