



國產IC開發套件 HUB 8735 示範案例

HUB 8735 Mask Detection





大綱

- 開發板HUB 8735 介紹
- AI Model 訓練
- 成果示範與展示



HUB 8735 介紹

HUB 8735 介紹

HUB 8735



晶片原廠
瑞昱半導體

晶片採用
Ameba RTL8735

【開發板特點】

- 兼容Arduino開發特性
- 具備多功能影像處理的高度集成模組
- 內置NPU AI 運算引擎加速處理AI模型
- 802.11 a/b/g/n 雙頻Wi-Fi與BLE低耗電藍牙傳輸
- 可廣泛應用於各種結合影像識別或AI運算之物聯網場域



經濟部工業局廣告

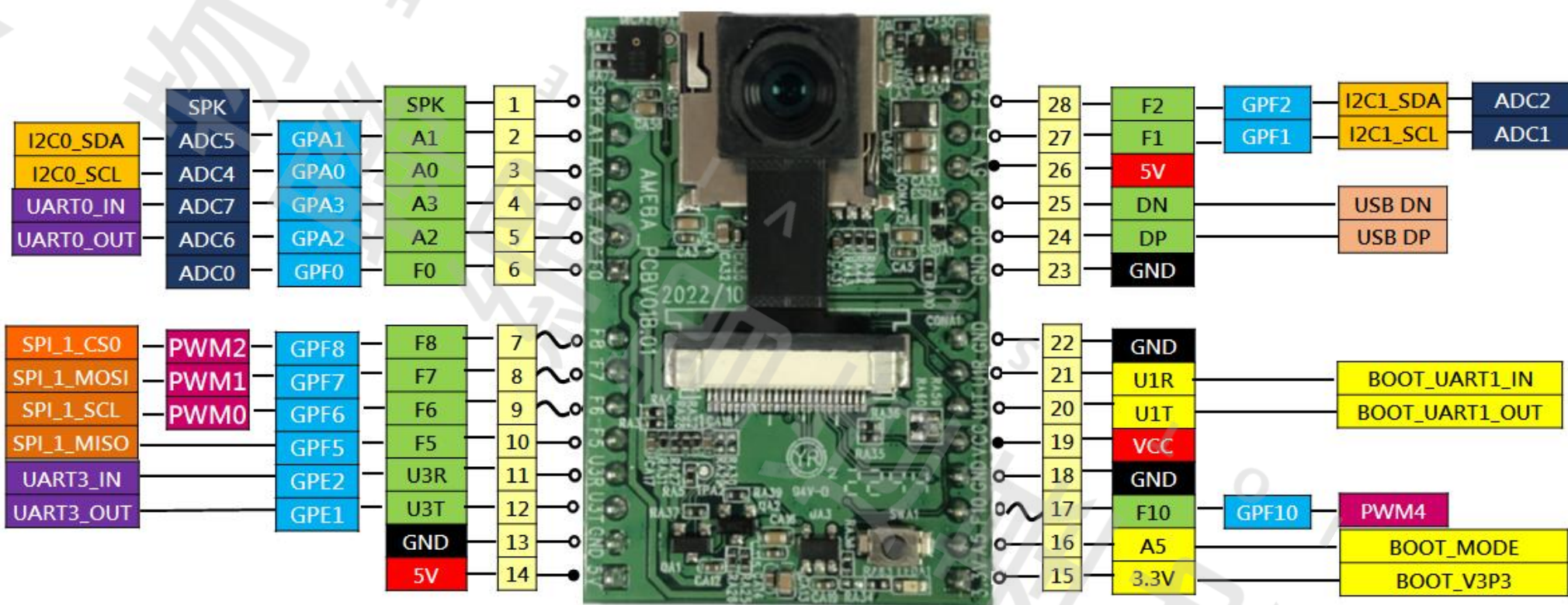
HUB 8735硬體規格

| 功能 | 描述 |
|----------------------|--|
| 處理器 MCU | RTL8735B AIoT Chips |
| 影像輸入 Video Input | Full HD 1080P CMOS |
| 語音輸入 Audio Input | Built-in MIC, audio input supported |
| 儲存裝置 Storage | Support external SD card |
| 無線連接 Connectivity | 802.11 a/b/g/n Wi-Fi 2.4GHz & 5GHz Bluetooth BLE RTSP (Real-Time Streaming Protocol) |
| 影像壓縮 Codec | H.264/265 |
| AI Models | Provides multiple pre-trained AI models, yolov3-tiny, yolov4-tiny, yolov7-tiny |
| I/O 界面 Interface | GPIOx15, I2Cx2, UARTx2, SPIx1, PWMx4, ADCx7 |

HUB 8735 介紹

HUB 8735 腳位圖

| |
|--------------------------------------|
| POWER |
| GROUND |
| PIN INDEX |
| PIN NAME |
| GPIO |
| ANALOG |
| PWM |
| UART |
| SPI |
| I2C |
| USB |
| CONTROL |
| 5V tolerant |
| Not 5V tolerant |
| PIN Mux |
| Max ±16mA per pin, ±4mA recommend |





HUB 8735 特色

- 多組硬體介面方便擴充
- 市場上少數結合Camera的 IoT模組
- 市場上少數具備AI算力IoT模組
- 台灣國產網通晶片，導入產品沒有疑慮
- 支持Arduino原生開發環境
- 透過雲端載入不同AI 模型
- AI功能無限擴充，第一階段釋出AI功能為物件辨識、人臉辨識、聲音種類辨識

其他資訊: https://www.ideas-hatch.com/mem_evb.jsp



AI Model 訓練

Dataset 下載

- **Original Dataset:**

- ◆ <https://www.kaggle.com/datasets/andrewmvd/face-mask-detection/discussion>

- **Augmented Dataset:**

- ◆ <https://universe.roboflow.com/iii-uimdg/mask-detection-vdfr3>



Google Drive

在Google Drive 建立名為maskdetection的 folder (名稱可以自定義) ,
在其內再建立images & labels,
並將dataset內的所有image & labels 放至對應資料夾

也將model 的一些設定檔案一併放在資料夾內

My Drive > maskdetection ▾

Type ▾

People ▾

Modified ▾

Name ↓

results

labels

images

val.txt

trainval.txt

train.txt

test.txt

README.roboflow.txt

README.dataset.txt

my_yolov4-tiny.cfg

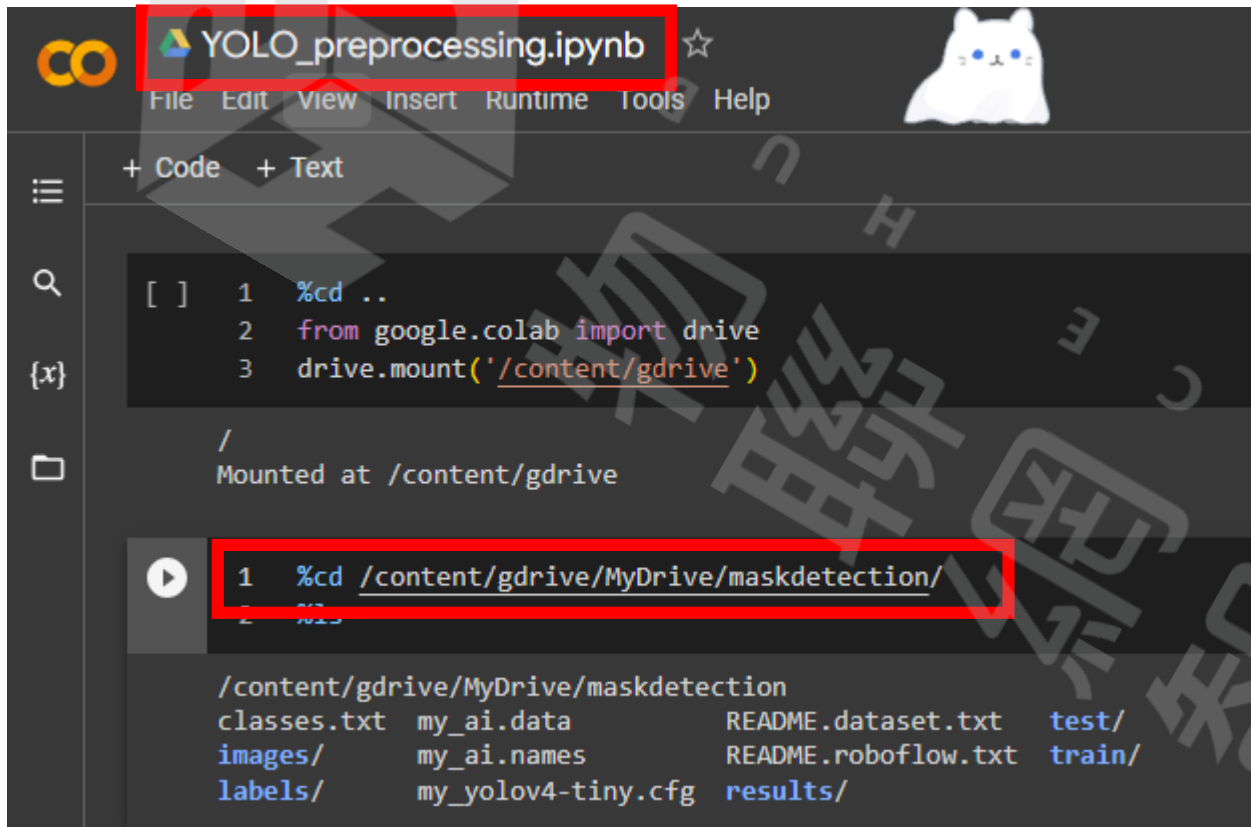
my_ai.names

my_ai.data

classes.txt

all_train.txt

Preprocess



YOLO_preprocessing.ipynb

```
File Edit View Insert Runtime Tools Help
```

+ Code + Text

```
[ ] 1 %cd ..
    2 from google.colab import drive
    3 drive.mount('/content/gdrive')
```

/

Mounted at /content/gdrive

```
1 %cd /content/gdrive/MyDrive/maskdetection/
```

/content/gdrive/MyDrive/maskdetection

| | | | |
|-------------|--------------------|---------------------|--------|
| classes.txt | my_ai.data | README.dataset.txt | test/ |
| images/ | my_ai.names | README.roboflow.txt | train/ |
| labels/ | my_yolov4-tiny.cfg | results/ | |

將資料集分成train & test



```
1 import os
2 import random
3
4 trainval_percent = 0.1
5 train_percent = 0.05
6 txtfilepath = './labels/'
7 imgfilepath = './images/'
8 total_txt = os.listdir(txtfilepath)
9
10 num = len(total_txt)
11 list = range(num)
12 tv = int(num * trainval_percent)
13 tr = int(tv * train_percent)
14 trainval = random.sample(list, tv)
15 train = random.sample(trainval, tr)
16
17 ftrainval = open('trainval.txt', 'w')
18 ftest = open('test.txt', 'w')
19 ftrain = open('train.txt', 'w')
20 fval = open('val.txt', 'w')
21
22 for i in list:
23     name = imgfilepath + total_txt[i][:4] + '.jpg\n'
24     if i in trainval:
25         ftrainval.write(name)
26         if i in train:
27             ftrain.write(name)
28         else:
29             fval.write(name)
30     else:
31         ftest.write(name)
32
33 ftrainval.close()
34 ftrain.close()
35 fval.close()
36 ftest.close()
```

Image 資料夾
Labels 資料夾



After Processing

My Drive > maskdetection ▾

Type ▾

People ▾

Modified ▾

Name ↓

results

labels

images

val.txt

trainval.txt

train.txt

test.txt

README.roboflow.txt

README.dataset.txt

my_yolov4-tiny.cfg

my_ai.names

my_ai.data

classes.txt

all_train.txt

會產生這些 .txt 檔案



Start Setup for Training

The screenshot shows a Google Colab notebook interface. The top bar includes the Colab logo, the notebook title 'mask_detection.ipynb', and a star icon. Below the title bar is a menu with 'File', 'Edit', 'View', 'Insert', 'Runtime', 'Tools', and 'Help'. The left sidebar contains a 'Table of contents' with the following items:

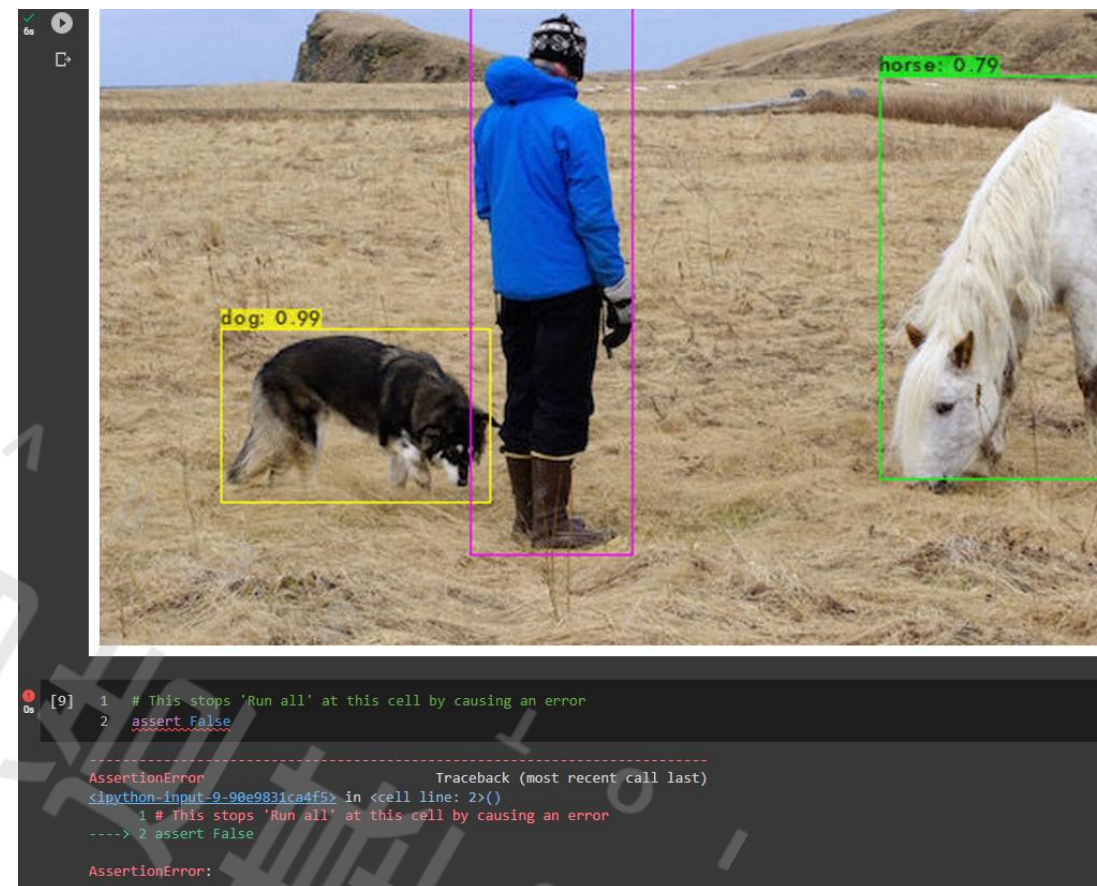
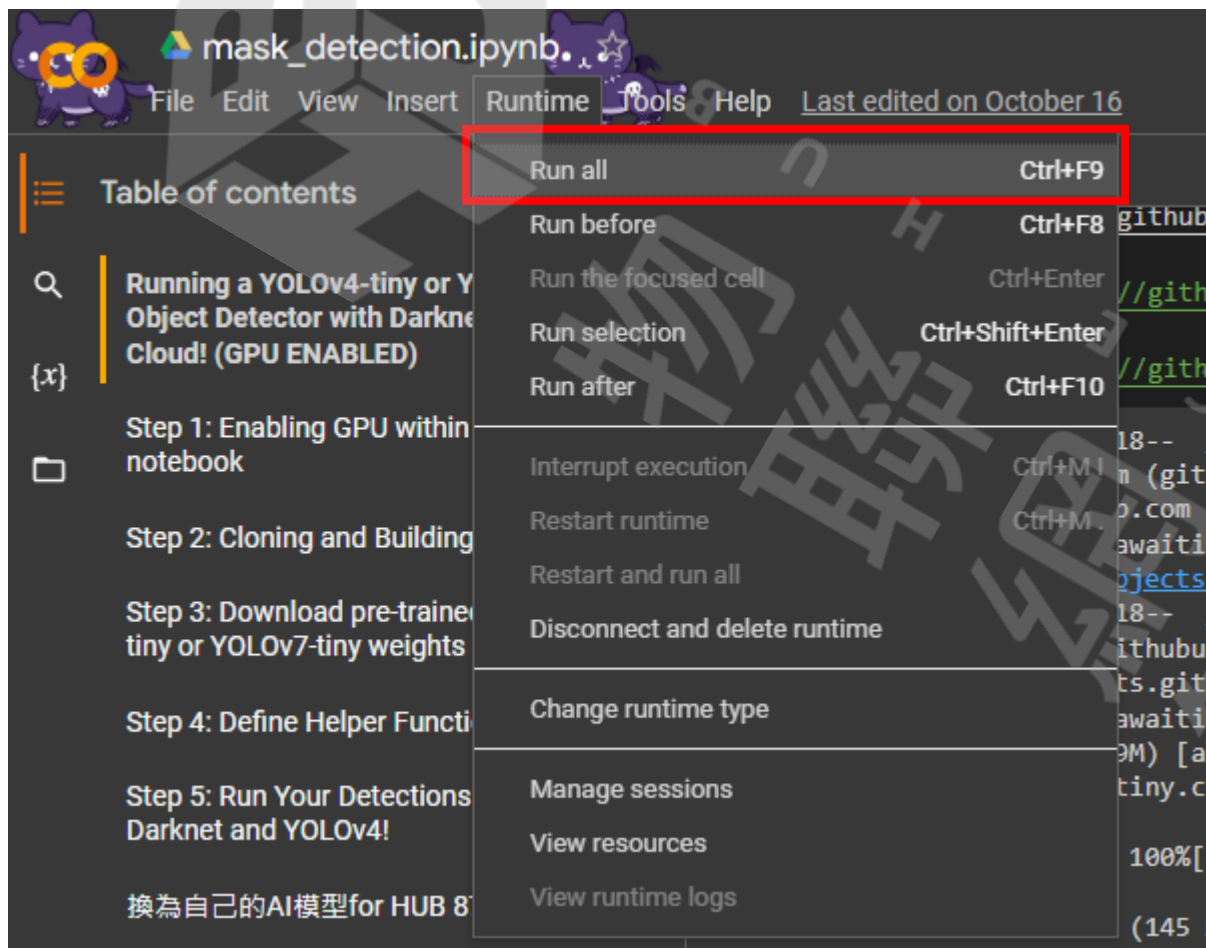
- Running a YOLOv4-tiny or YOLOv7-tiny Object Detector with Darknet in the Cloud! (GPU ENABLED)
- Step 1: Enabling GPU within your notebook
- Step 2: Cloning and Building Darknet
- Step 3: Download pre-trained YOLOv4-tiny or YOLOv7-tiny weights
- Step 4: Define Helper Functions
- Step 5: Run Your Detections with Darknet and YOLOv4!

The main content area displays the title 'Running a YOLOv4-tiny or YOLOv7-tiny Object Detector with Darknet in the Cloud! (GPU ENABLED)' and a subtitle 'This tutorial will help you build YOLOv4 easily in the cloud with GPU enabled so that you can run object detections in milliseconds!'. Below this, the first step is titled 'Step 1: Enabling GPU within your notebook'. The text explains that GPU acceleration is needed for faster processing. The steps are:

- Click **Edit** at top left of your notebook
- Click **Notebook Settings** within dropdown

Two inset images illustrate these steps. The first inset shows the 'Edit' button in the top left corner of the notebook interface, highlighted with a red box and a red arrow. The second inset shows the 'Runtime' menu dropdown, with 'Notebook Settings' highlighted.

Start Setup for Training



看到這張圖像就表示darknet環境建立好了

開始訓練

訓練自己的AI模型

my_yolov4-tiny

```
[ ] 1 from google.colab import drive  
    2 drive.mount('/content/gdrive')
```

連接Drive

Mounted at /content/gdrive

```
[ ] 1 #/content/gdrive/MyDrive/my_dataset  
    2 !cp -r /content/gdrive/MyDrive/maskdetection/* /content/darknet/
```

將資料集複製到darknet

```
[ ] 1 !cp -r /content/darknet/labels/* /content/darknet/images/
```

```
[ ] 1 %mkdir /content/gdrive/MyDrive/maskdetection/results
```

建立results folder 用以存weights

開始訓練

```

1 %cd /content/darknet
2 %ls

/content/darknet
3rdparty/
all_train.txt
backup/
build/
build.ps1*
cfg/
classes.txt
cmake/
CMakeLists.txt
darknet*
DarknetConfig.cmake.in
darknet_images.py
darknet.py
darknet_video.py
data/
docker-compose.yml
Dockerfile.cpu
Dockerfile.gpu
images/
image_yolov3.sh*
image_yolov4.sh*
include/
json_mpeg_streams.sh*
labels/
LICENSE
Makefile
my_ai.data
my_ai.names
my_yolov4-tiny.cfg
net_cam_v3.sh*
net_cam_v4.sh*
obj/
package.xml
README.dataset.txt
README.md
README.roboflow.txt
results/
scripts/
src/
test/
test.txt
train/
train.txt
trainval.txt
val.txt
vcpkg.json
vcpkg.json.opencv23
video_yolov3.sh*
video_yolov4.sh*
yolov4-tiny.conv.29
yolov4-tiny.weights

```

Darknet裡應出現這些檔案

開始訓練

開始訓練

指定.data, .cfg, 預訓練參數檔並開始訓練，如需記錄平均精確度mAP(mean average precisions)可加上參數 -map。如果訓練過程很容易無故中止則建議移除參數 -map。

訓練期間會記錄Loss並繪成圖表(darknet/chart_my_yolov4-tiny-custom.png)，可隨時重新開啟觀察訓練成果。

每隔1000次會自動備份一次權重檔到雲端硬碟/my_drive/yolov4-tiny下(根據my_obj.data中backup設定值)，檔名為my_yolov4-tiny_x000.weights (x為1~n)。

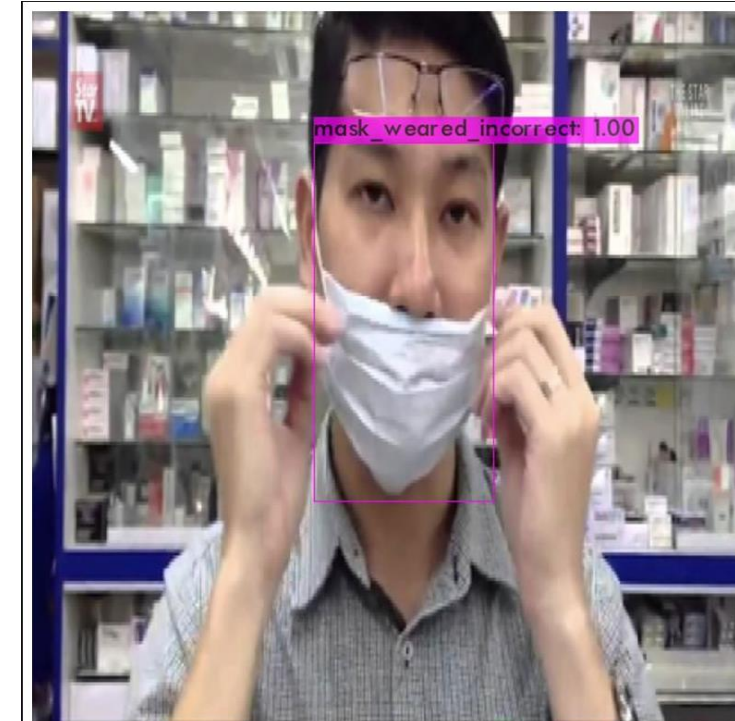
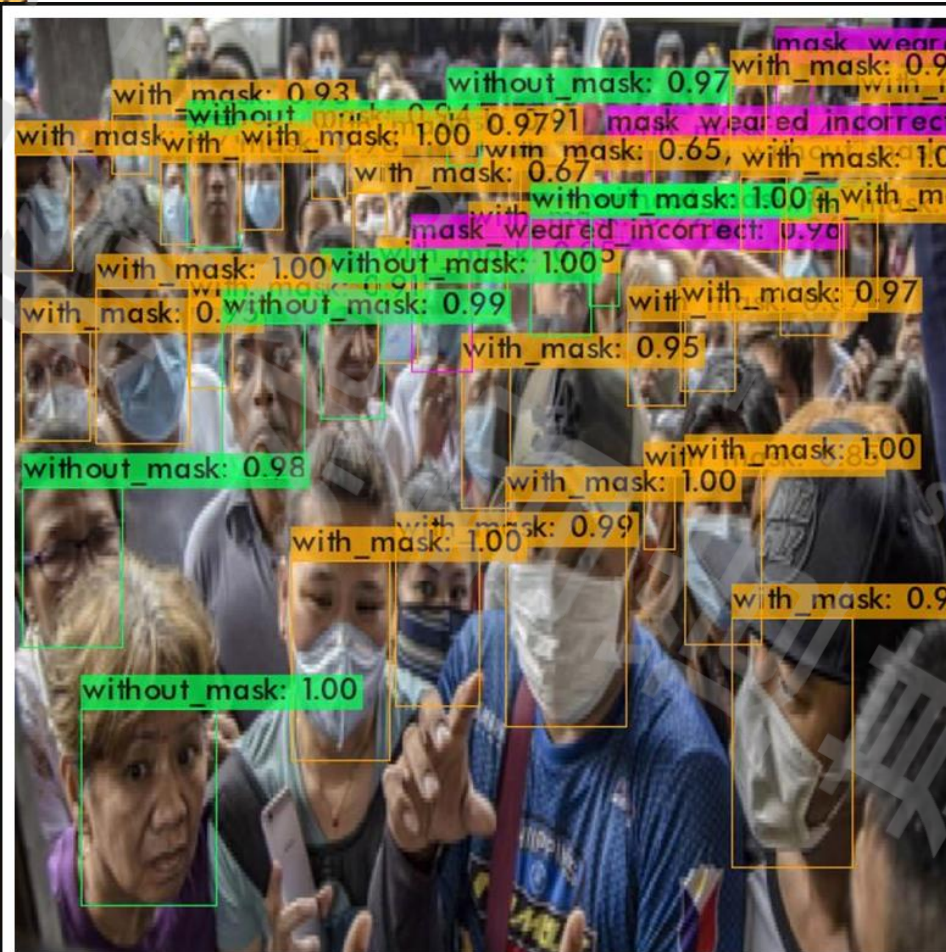
訓練期間會自動產生my_yolov4-tiny_best.weights和my_yolov4-tiny_last.weights，完成訓練會產生my_yolov4-tiny_final.weights。

這裡要注意雲端硬體要有足夠空間存放，否則空間不足時就無法備份權重值。

```
1  !./darknet detector train my_ai.data my_yolov4-tiny.cfg yolov4-tiny.weights -dont_show -map
2
```

!./darknet detector **train** <DATA FILE> <CONFIG FILE> <WEIGHTS> -dont_show -map

訓練結果



轉換模型

| | | |
|-----------------------------|-------------------|----------------------|
| images | 8/9/2023 10:07 AM | File folder |
| labels | 8/9/2023 10:07 AM | File folder |
| nb files | 8/11/2023 9:39 AM | File folder |
| classes | 8/9/2023 2:05 AM | Text Document |
| my_ai | 8/1/2023 4:50 PM | DATA File |
| my_ai | 8/1/2023 3:07 PM | NAMES File |
| my_yolov4-tiny | 8/11/2023 9:35 AM | Configuration Sou... |
| my_yolov4-tiny_1200.weights | 8/9/2023 5:16 PM | WEIGHTS File |
| my_yolov4-tiny_1300.weights | 8/9/2023 3:28 PM | WEIGHTS File |
| my_yolov4-tiny_2200.weights | 8/11/2023 9:35 AM | WEIGHTS File |
| my_yolov4-tiny_best.weights | 8/9/2023 2:47 PM | WEIGHTS File |

將my_yolov4-tiny.cfg & 訓練好的weights 進行壓縮成zip

| | | | |
|-----------------|-------------------|---------------------|-----------|
| ringgits_yolov4 | 8/11/2023 9:36 AM | Compressed (zipp... | 21,363 KB |
| test | 8/1/2023 12:10 PM | JPG File | 59 KB |
| test1 | 8/9/2023 2:05 AM | JPG File | 239 KB |



注意

Please Log in to access the page.

目前線上工具僅支援yolov3-tiny、yolov4-tiny、yolov7-tiny、mobilefacenet和scrfd模型。

應用AI模型轉換對於yolo-tiny需要上傳".cfg"、".weights"，對於scrfd/mobilefacenet需要上傳".pt"或".onnx"文件，量化圖像最少上傳1張，最多可上傳10張。

AI模型轉換完成後，將通過電子郵件發送下載連結，更多信息請參考自定義AI模型安裝指南在<https://www.amebaiot.com/zh/ameba-arduino-summary/>。

請參考 AMB82 MobileFaceNet Convert To ONNX 在 <https://www.amebaiot.com/zh/amebapro2-mobilefacenet-convert-to-onnx/>。

請參考 AMB82 SCRFD Convert To ONNX 在 <https://www.amebaiot.com/zh/amebapro2-scrfd-convert-to-onnx/>。

Go to:

<https://www.amebaiot.com/zh/amebapro2-ai-convert-model/>

E-mail (required, After submission, we will send out download link to you through email.)

Confirm E-mail (required)



Submit and Check Your Email

E-mail (required, After submission, we will send out download link to you through email.)

abc@gmail.com

Confirm E-mail (required)

abc@gmail.com

Model (required, YOLO-TINY or SCRFD or MobileFaceNet)

YOLO-TINY

Quantize Type (required, UINT8 or INT16)

UINT8

Upload zip file including a cfg file and a weights file(required, please upload the folder or compressed file contained the ".cfg" and ".weights" files, all named in English, limit:35MB)

ringgits_yolov4.zip

Upload one jpg file (required, limit:1MB)

test1.jpg

Upload jpg files (option, limit:1MB)

No file chosen

AmebaPro2 AI convert Inbox x



root <root@amebaiot.com>

 to me ▾

 Filipino ▾

> English ▾

[Translate message](#)

Please download your file: https://www.amebaiot.com/wp-content/uploads/MQTT/rodneytai97/network_binary.nb

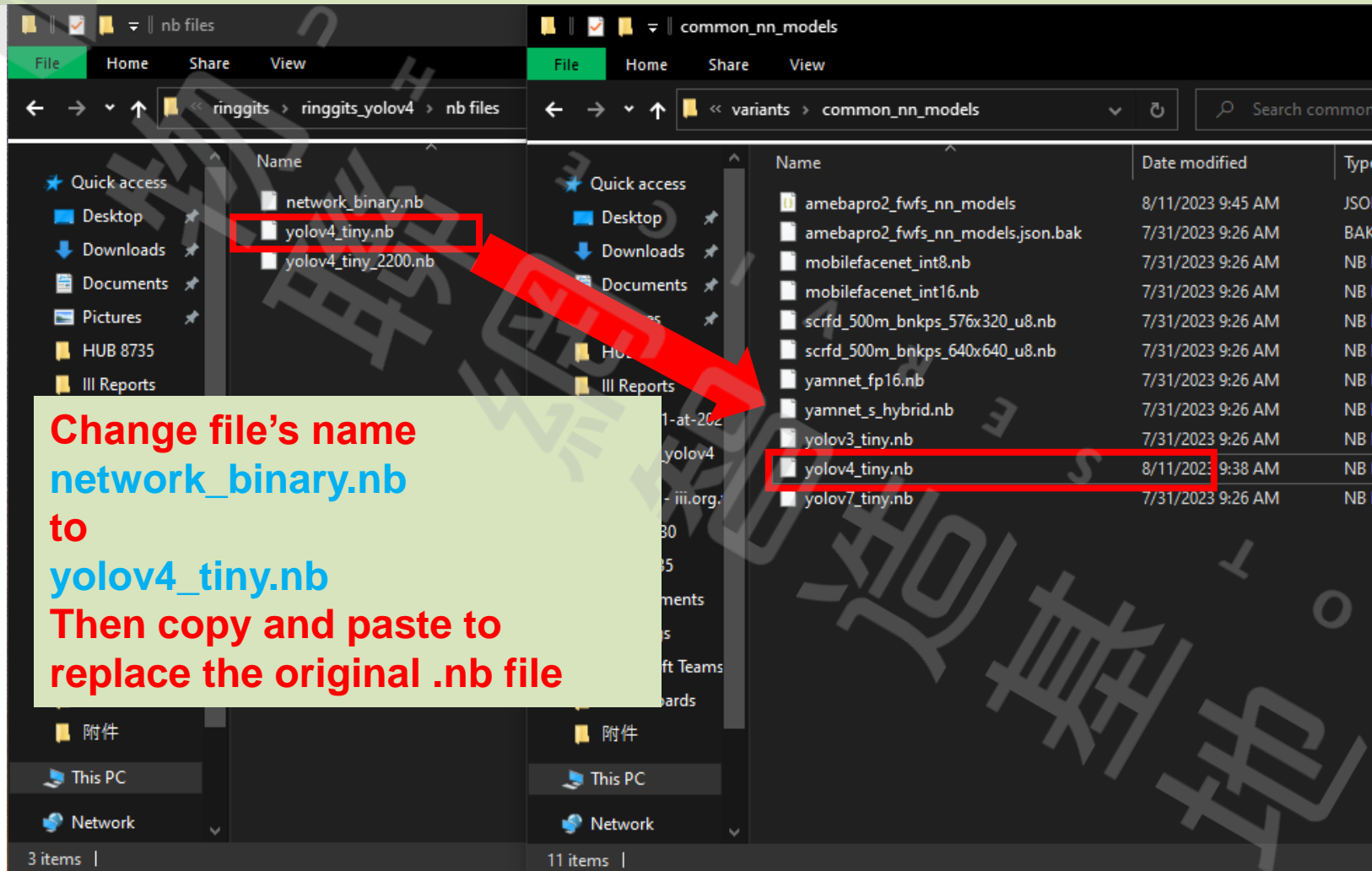
 Reply

 Forward

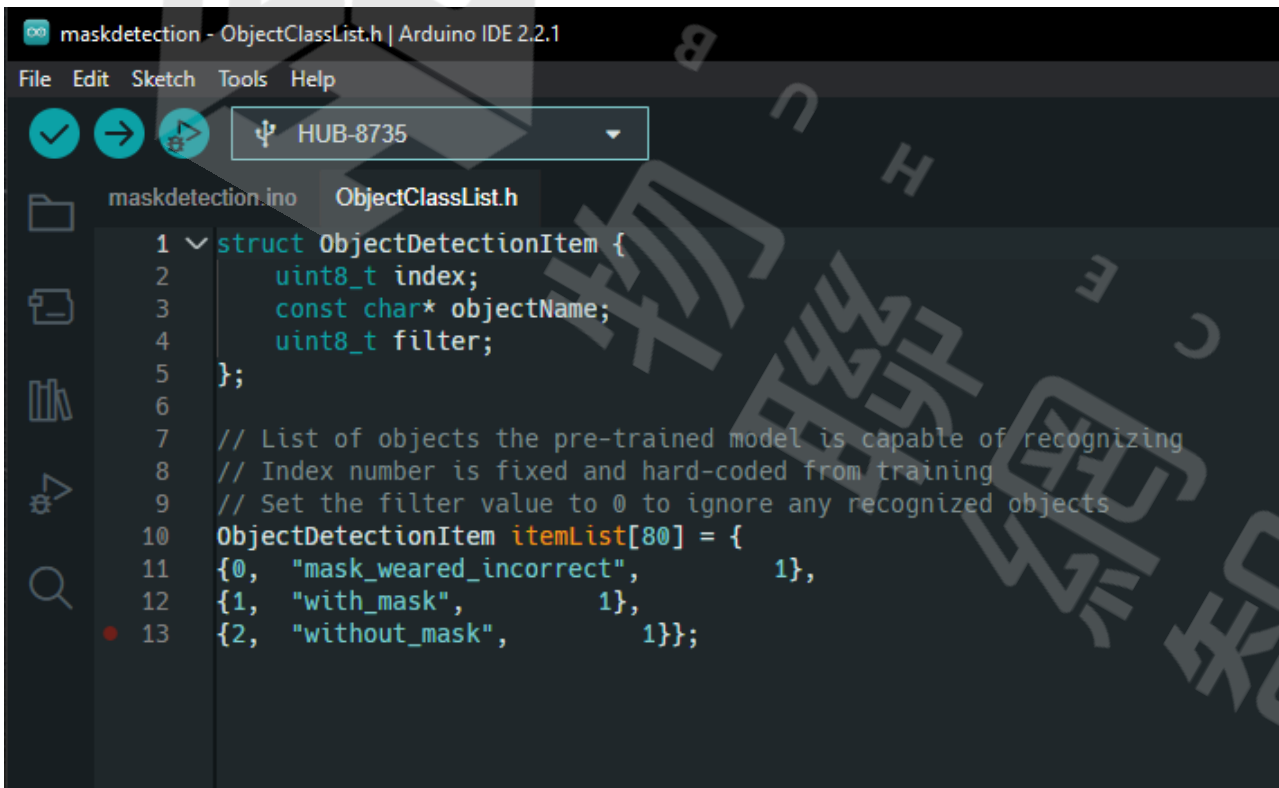
替換Network Binary(.nb) file

Go to:

C:\Users\<USERNAME>\AppData\Local\Arduino15\packages\realtek\hardware\AmebaPro2\
<BUILD_VERSION>\variants\common_nn_models



開啟範例程式

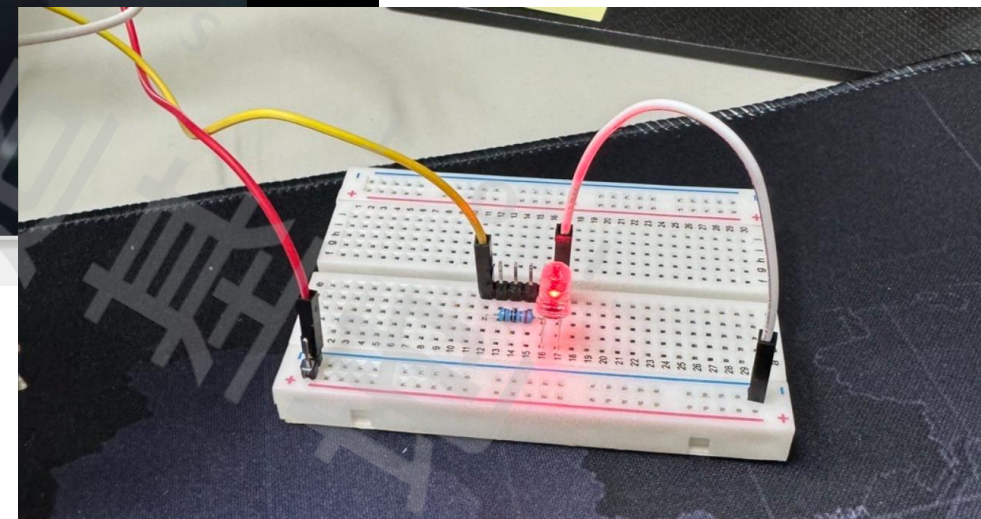


```
maskdetection - ObjectClassList.h | Arduino IDE 2.2.1
File Edit Sketch Tools Help
ψ HUB-8735
maskdetection.ino ObjectClassList.h
1 struct ObjectDetectionItem {
2   uint8_t index;
3   const char* objectName;
4   uint8_t filter;
5 };
6
7 // List of objects the pre-trained model is capable of recognizing
8 // Index number is fixed and hard-coded from training
9 // Set the filter value to 0 to ignore any recognized objects
10 ObjectDetectionItem itemList[80] = {
11 {0, "mask_wearred_incorrect", 1},
12 {1, "with_mask", 1},
13 {2, "without_mask", 1}};
```

```
125 void ledAlert()
126 {
127   unsigned long currentMillis = millis();
128   if (currentMillis - previousMillis >= interval)
129   {
130     // It's been at least 0.9 second since the last execution
131     previousMillis = currentMillis; // Save the current time
132
133     for (int i = 0; i < 3; i++)
134     {
135       digitalWrite(LED_PIN, HIGH);
136       delay(75);
137       digitalWrite(LED_PIN, LOW);
138       delay(75);
139     }
140   }
141 }
```

LED 正極連接至 HUB 8735的 F6腳位
LED亮起function ,
 辨識到沒戴口罩/不正確戴法時會亮起 , 以作為警示的功能

成果展示





Thank you



物聯網智造基地

IOT SERVICE HUB