國產IC開發套件 HUB 8735 手勢辨識居家智慧

C





- ➤ IC開發板:HUB-8735
- ▶ 案例/功能說明:
 - 手勢辨識居家智慧系統·能提供特定族群解決生活上的不便
 - 透過 HUB-8735 開發板連接到智能家居網路的開關或電器,再藉由鏡頭偵測手勢,即可控 制開關燈或是電器用品
- ▶ 搭配模組:
 - RGB LED 燈模組
- ▶ 其他補充:
 - 採用YOLOv4 進行AI訓練





HUB 8735介紹

HUB 8735 特色

- ▶ 多功能影像處理
- ▶ 內建的NPU加速處理AI模型
- ▶ 支援Wi-Fi和藍芽傳輸
- ▶ 內置多款預訓練的AI模型
- > 適用於多個領域
- > 小尺寸設計可整合到產品設計中





HUB 8735 規格

功能	8
處理器	RTL8735B AloT Chips
影像輸入	Full HD 1080P CMOS
語音輸入	內建數位麥克風
儲存裝置	支援SD記憶卡
無線支援	Wi-Fi 2.4 GHz / 5 GHz Bluetooth BLE 4.2 無線影像串流
影像壓縮	H.264/265
AI處理	提供多種pre-trainedAl models
I/O 界面	依照開發者需求擴充功能 1.Speaker 語音輸出 2.IMU Sensor 3.擴充溫度、溼度、震動 Sensor。

HUB 8735 接腳圖

DESI	GN & DE IN TAIWAN					HUB 8735	
AOUT 1/0 1/0 1/0 1/0 1/0	UARTO_IN UARTO_OUT	I2C0_SDA I2C0_SCL	ADC5 ADC4 ADC7 ADC6 ADC0	GPA1 GPA0 GPA3 GPA2 GPF0	SPK A1 A0 A3 A2 F0	F2 GPF2 I2C1_SDA ADC2 F1 GPF1 I2C1_SCL ADC1 SV DN DP GND	I/O I/O POW USB USB POW
/0 /0 /0 /0 /0 /0 POW POW	UART3_IN UART3_OUT	SPI_1_CS0 SPI_1_MOSI SPI_1_SCL SPI_1_MISO	PWM2 PWM1 PWM0	GPF8 GPF7 GPF6 GPF5 GPE2 GPE1	43.9mm F8 F7 F6 F5 U3R U3T GND 5V	GND UIR UIT VCC GND F10 GPF10 PWM4 A5 3.3V	POW 1/0 1/0 P_OU POW 1/0 BOO

30.5mm



開發環境

- Step.1:至 Ardunio 官網下載 Arduino IDE 1.8.19 之後的版本 https://www.arduino.cc/en/software
- Step.2:開啟Arduino IDE,選取檔案>偏好設定>在其他開發版版 管理員網址貼上SDK地址

<u>https://github.com/ideashatch/HUB-</u> 8735/raw/main/amebapro2_arduino/Arduino_package/ideasHatch.json

	檔案	編輯	Sketch	工具	說明(H)	
	新增	sketch	1	Ctr	l + N	T	;
	新增	雲 sket	ch Al	t + Ctr	l + N		
	開啟.			Ctr	l + 0		
	開啟i	最近的				►	nt N
	Sketo	hboo	k			►	e it
	範例					►	onf:
	翩閉			Ctrl	+ W		e it tri
	Save			Cti	rl + S		nt v
Step.2	另存新	新檔	Ctr	l + Shif	t + S		onf:
	喜好詞	設定…		Ctrl +	逗號		e 11 onf:
	進階					►	e i
	結束((Q)		Ctr	l + Q		nt v
		1	65			С	onf
		1	66		}	els	e i

t i	喜好設定 × ×
i i	設定 網路
	sketchbook 位置:
	c:\Users\Peter\Documents\Arduino
	■ 顯示 sketch 裏面的檔案
	編輯器字體大小: 14
	使用者介面縮放比例: 🛛 🖬 100 %
	色彩佈景主題:
	編輯器語言: 中文(繁體) (Reload required)
	在以下階段顯示詳細輸出 ■ 編譯 ■ 上傳
	編譯器警告 無・・
	■上傳後驗證程式碼
	✓ 自動儲存(U)
	■ 編輯希状迷迷論 甘他問發時等週号编批· https://aithub.com/idoachatch/UUB 8735/raw/main/amohanro2_arduino/Arduino_packag
	会他曲弦版音注受副组。 https://gittub.com/actandici//105-07-0/rawintain/anti-bapioz_artain//wain/o_packag
	(取消) 確定(0)

開發環境

Step. 3:再從工具> HUB 8735的開發版>開發板管理員中搜尋: HUB 8735, 並安裝最新版的 HUB 8735 板子

• Step. 4:至以下網址下載 VLC Media Player

https://github.com/portapps/vlc-portable/releases

	工具 說明(H)						
	自動格式	Ctrl + T					
	sketch 歸檔						
	。 管理程式庫	Ctrl + Shift + I			57.57		
	序列埠監控窗	Ctrl + Shift + M		<pre>= value.toInt(); pr actContract(upl);</pre>			
	- 序列埠監控窗			<pre>input.startsWith(String("CONTRAST"))) {</pre>			
O (O				am.getContrast();			
Step.3				<pre>input.startsWith(String("SATURATION="))) { //alue = input substring(11)</pre>			_
	:"HUB-8735" 開發板			開發板管理員	Ctrl + Shift + B		\sim
	· · · · · · · · · · · · · · · · · · ·		►			►	
	o 取得開發板資訊			esp8266		•	
	🗸 🔹 * Auto Flash Mode: "Disable"			 ideasHatch Ameba Boards (32-bit Arm v8M @500MHz) 		•	
	* Erase All Flash Memory (16MB): "Disable"			<pre>= value.toInt();</pre>			
	* Standard Lib: "Arduino_STD_PRINTF"			am.setSharpness(val);			
	V * Upload Speed: "2000000"			<pre>input.startsWith(String("SHARPNESS"))) { am.getSharpness():</pre>			
	D 燒錄 Bootloader Ortenta watchine control 169	SULT	ng	<pre>input.startsWith(String("LDC="))) { value = input.substring(4);</pre>			







• 將HUB 8735與USB to TTL進行連接 連接方式如下圖所示

DESIGN & MADE IN TAIWAN HUB 8735 I2C1_SDA ADC2 1/0 AOUT . GPF1 I2C1_SCL ADC1 1/0 20 ADC5 1/0 12C0 SDA 5V POW 12C0_SCL ADC4 1/0 50 . • USB UARTO_IN ADC7 1/0 USB ADC6 UARTO_OUT G 📀 1/0 GND POW ADC0 GPF0 0 1/0 43.9mm -POW 1/0 SPI 1 CS0 PWM2 PWM1 1/0 0 1/0 SPI 1 MOSI 1/0 PWM0 1/0 SPI_1_SCL 30 • • VCC P_OUT 1/0 SPI_1_MISO • GND POW GPE2 1/0 UART3_IN 1/0 PWM4 1/0 UART3_OUT GPE1 U3T 0 A5 BOOT GND POW 0 BOOT POW •

30.5mm

USB to TTL

<u>ب</u>



• Step. 1:開啟程式 範例>AmebaNN>ObjectDetectionCallback

• Step. 2:選擇開發板及序列埠

檔案(F) 動作(Δ) 檢視(V) 說明(H)	IDjectDetectionCaliback Arduino IDF 2.3.2 道語 Sketch 工具 約用(小)
	→ ● ● 自動格式 Ctrl + T sketch 歸柵
> ■ 安全性裝置	ObjectDete 11F 管理程式庫 Ctrl + Shift + I
> 🏣 系統裝置	119 116 序列埠監控窗 Ctrl + Shift + M
> 🚺 其他裝置	117 序列埠監控资
> ⑨ 相機	118 119 - 动赠更新
> 💵 音效、視訊及遊戲控制器	120 上傳SSL根憑証 うてらり、 2
> 💵 音訊處理物件 (APO)	121 122 : "HUB-8735" 開發板 → 開發板管理員 Ctrl + Shift + B
> 👖 音訊輸入與輸出	122 建接埠::"COM9" Arduino AVR Boards ▶
> 🔲 處理器	124 V m-#∃BB23x⊂≫en esp8266
> 📑 軟體元件	125 可從裝置管理員中查出 COM 編號 ● ideasHatch Ameba Boards (32-bit Arm v8M @500MHz)
	127 } * Erase All Flash Memory (16MB): "Disable" HUB-8735_ultra
> 🖞 通用序列匯流排控制器	128 129 / *Standard Lib: "Arduino_STD_PRINTF" iect detection results
✓ ₩ 建按焊(COMI和LPT)	130 v *Upload Speed: "2000000"
■ USB-SERIAL CH340 (COM9) ■ 透過鼓牙連结的標準 (COM3)	131 { 病發 Bootloader 132 ····································
₩ 透過藍牙連結的標準序列 (COM4)	

- > 🎽 韌體
- > 🕕 滑鼠及其他指櫄裝置
- 《금 품과

Step. 3: 輸入Wi-Fi SSID 及密碼

Step. 4:插入BOOT_MODE跳腺並按下Reset鈕



Step.5

🧧 Objec	ObjectDetectionCallback Arduino IDE 2.3.2						
檔案編	輯 Sketch	工具 說明(H)					
		₽ HUB-8735 👻					
r l	CojeciDeia	CiunCallback.ino ObjectClassList.h					
	115	OSD.contigVideo(CHANNEL, contig);					
<u> </u>	116	OSD.begin();					
[_]	117						
	118	<pre>//configCam.setGrayMode(1);</pre>					
mL.	119	3					
	120	<pre>pinMode(LED_BUILTIN, OUTPUT);</pre>					
//	121	<pre>pinMode(9, OUTPUT);</pre>					
	122	}					
3	123						
	124	void loop()					
\bigcirc	125	{					
	126						
	127	}					
	128						
	129	// User callback function for post pr					
	130	<pre>void ODPostProcess(std::vector<object< pre=""></object<></pre>					
	131						
	132	<pre>uint16_t im_h = config.height();</pre>					
	4.2.2	and a local distance of the second dist. (As a					

ObjectDetectionCallback | Arduino IDE 2.3.2

- Śww Step. 6: 燒入完成之後拔掉BOOT_MODE跳腺
 - Step. 7:開啟序列埠監控視窗
 - Step. 8:設定鮑率為115200
 - Step. 9:按下Reset鈕
 - Step. 10:查看IP位址

152 // Multiply with RTSP resolution to get coordinates in pixels
153 int xmin = (int)(item.xMin() * im_w);
154 int xmax = (int)(item.xMax() * im_w);
155 int ymin = (int)(item.yMin() * im_h);
156 int ymin = (int)(item.yMin() * im_h);
157 int ymin = (int)(item.yMin() * im_h);
158 int ymin = (int)(item.yMin() * im_h);
159 int ymin = (int)(item.yMin() * im_h);
150 int ymin = (int)(item.yMin() * im_h);
150 int ymin = (int)(item.yMin() * im_h);
151 int ymin = (int)(item.yMin() * im_h);
152 int ymin = (int)(item.yMin() * im_h);
153 int ymin = (int)(item.yMin() * im_h);
155 int ymin = (int)(item.yMin() * im_h);
1

輸出 序列埠監控窗

Sketch uses 14995456 bytes (89%) of program storage space. Maximum is 16777216 bytes

Tota YOLC Netw

Tota YOLO Netw

Tota

```
Start Upload Flash
```

Uploading......upload success

Step.6 燒入成功

○↑ 缶井 檔案 編輯 Sketch 工具 說明(H)	Step.7
	序列埠監 ช窗 사 📢
ObjectDetectionCallback.ino ObjectClassList.h	序列埠監
41 CameraSetting configCam;	
<pre>to get coordinates in pixels ietting config(VIDEO_FHD, 30, VIDEO_H264, 0); m_w); ietting configNN(NNULTDTH_NNULFICHT_10_VIDEO_H</pre>	
"" ":ctDetection ObjDet;	NGD, 0),
ace. Maximum is 16777216 bytes.	
IIO VideoStreamer(1, 1); IIO videoStreamerNN(1, 1);	2
	Ctop 0
	Step.8 ≽ ⊘
(按 Enter 鍵將訊息發送到 COM9 上的 HUB-8735) / 換行 / 換行	▼ 115200鮑率
1 number of objects detected = 0	
v4t tick[242]	
ork URL for RTSP Streaming: rtsp://192.168.227.179:554	
1 number of objects detected = 0	
v4t tick[233] prk URL for RTSP Streaming _ rtsp://192.168.227.179:554 序列埠盤	控視窗
1 number of objects detected = 0	



• Step. 11:打開VLC Media Player >媒體>串流

• Step. 12: 選取網路> 並於網路通訊協定貼上IP 位址

🛓 VLC 媒體播放器



🛓 開啟媒體			- 🗆 X
 ▶ 檔案(F) ◆ う →網路通訊協定 	光碟(D) ♥網路(N) ♥ 擷取裝置(D)		Step.12
	月朝和八海马山 rtsp://192.168.227.179:554		~
5	http://www.example.com/stream.avi htp://wit254 mms://mms.examples.com/stream.asx rtsp://server.example.org:8080/test.sdp http://www.yourtube.com/watch?v=gg64x		以物件領測為範例
	Sm.		S
☑顯示更多選項()	M)		
快取 1000 3	毫秒 <u>-</u>	開始時間	00H:00m:00s.000 ÷
□同步播放另一	這邊可以調整畫面更	新頻率間	00H:00m:00s.000 😫
MRL	rtsp://192.168.227.179:554		
編輯選項	:network-caching=1000		
			串流(S) → 取消(C)

燒錄HUB 8735-以物件偵測為範例 播放後就會開始撥放串流,您可以在播放器看到偵測到的物件決策框



若要關閉特定物件,可到_OBJECTCLASSLIST_H_ 把該物件的1改為0



17

訓練自己的模型

我們的程式碼都在: <u>https://github.com/lalalin412/hub8735.git</u>

照片訓練及資料: https://drive.google.com/drive/folders/1MMyN57JvJl1ECvJT1zq97zCGAZI7YD5A?usp=drive_link



AMB82 AI 模型轉換

● HUB-8735支援以下模型轉換



Please Log in to access the page.

目前支援線上轉換的型號請參考下表。

Models	Basic functions	Requires files
yolov3-tiny, darknet	Object Detection	".cfg", ".weights"
yolov4-tiny, darknet	Object Detection	".cfg", ".weights"
yolov7-tiny, darknet	".cfg", ".weights"	
yolov7-tiny, pytorch	".pt"	
scrfd/mobilefacenet	Face Detection & Recognition	".pt" or ".onnx"
yamnet	".h5"	
CNN Gray/RGB	Image Classification	".h5"

收集資料集-本範例的資料集共有6個手勢(數字0~5),全部有 1360張圖片



• 於以下網址下載labellmg

https://github.com/HumanSignal/labelImg/releases

• 解壓縮後進入data資料編輯類別



Step. 1:點擊資料夾內labellmg.exe

- Step. 2:選取圖片資料夾
- Step. 3: 選取txt存放目錄
- Step. 4:按下去直到變成
 YOLO為止
- Step. 5:畫框
- Step. 6:儲存資料



回到txt的存放目錄,就可以發現classes.txt 及剛剛儲存的txt檔,會與圖片檔同名,txt裡存著bounding box的資訊。

	classes.txt	0 (13) +-+	classes tyt	· · · · · ·	0 (13) txt	×
檔案	編輯檢視	楼 安	Classes.txt		0 (13).001	^
0 1 2 3 4 5			.481275 0. 第一個為 以編號為 第三個為 第五個為	514385 0.381696 0.508 急類別編號、O是第 るO るbounding box的 るbounding box的 るbounding box的 るbounding box的	8929 1種類別、所 中心點的×軸 中心點的y軸 width (寬度) height(高度)	

• 至google drive命名一個名為data資料夾

 在data資料夾內命名一個為JEPGImages資料夾,並將 你的圖片存入資料夾內

首頁
 我的雲端硬碼

公 與我共用
 ③ 近期存取
 ☆ 已加星號
 ① 垃圾内容
 回 垃圾桶
 ○ 儲存空間

 在data資料夾內命名一個為labels資料夾,並將你的 txt檔存入資料夾內

\bigtriangleup	雲端硬碟	Q. 在雲端硬碟中搜尋	手
+	新增	我的雲端硬碟 > data -	
â	首頁	類型 • 使用者 • 上次修改日期 •	
• 4	我的雲端硬碟	↔ 鍵盤快速鍵更新 Google 雲端硬碟將在 2024年8月1日 星期四更新鍵	盤快速鍵, 讓你透過首字母導覽功能
•	電腦		
		名稱	擁有者
ä	與戎共用		0.50
S	近期存取	JPEGImages	日 找
☆	已加星號	labels	8 我
!	垃圾内容		
	*注意		
			-

一定要按照資料集的格式存放,因為 darknet 原始碼中的介面定義就是需要 這樣放置和命名。

		Qŧ	E雲端硬碟中	中搜尋						
		我的	雲端硬	碟ゝ	data	>	JPEGIma	ges 🕶		
		類型	• 使	用者 🔹		口期	•			
		F 1:	鍵盤快速鍵	更新 Go	ogle 雲端硬碟	將在 2	024年8月1日 星期	月四更新鍵盤	計 決:	
		名稱	\uparrow						拥	
			0 (1).jpg						6	
			0 (2).jpg						¢	
			0 (3).jpg						6	
۵	雲端砌	更碟		Q 在	雲端硬碟中掛	夏尋			1	
7	新增			我的	雲端硬碟	>	data →	labels	•	
â	首頁			類型	• 使用	2 ·	上次修改日期	•		
, 6	我的雲端	硬碟		я÷	鍵盤快速鍵更新	昕 Goo	gle 雲端硬碟將在:	2024年8月1日	3 星期四更新	趕盤快速
• 🗖	電腦			An eres						200-0
ð	與我共用			名稱						摊月
0	近期存取				0 (30).txt					Θ
☆	已加星號	2			0 (29).txt					0
()	垃圾内容				0 (28).txt					Θ
	垃圾桶				0 (27).txt					8
- -	1887子全国	16 2 MP /	<i>P</i> #		0 (26).txt					6
存空	1927月重: C 間配額: 1	5 GB)	1818	-	0 (25) +++					



• 至Google Colab製作AI模型

https://colab.research.google.com/?hl=zh-tw



_		
Ż	https://colab.res	search.google.com
DTT	🛅 1 🛅 МСИТ 🛅	LINE 📋 論文 📁 3
歡	迎使用 Colaborator	ry 🔍
檔	案编辑檢視畫面插	· 入執行階段工具 言
	新増筆記本	文字
	開啟筆記本	Ctrl+O
	上傳筆記本	THE
	重新命名	
	在雪邊硬碑山儲左複木	i) (i
	將副本另存為 GitHub Gist	
	在 GitHub 中儲存副本	erat
	儲存	Ctrl+S ini/
	 修訂版本記錄	<u>ini /</u>
		<u> </u>
	列印	Ctrl+P
		如果你已經熟;

25



[™]● 打開LabelImg_to_YOLO_preprocessing · 按下執行階段>全部執行



 執行完成後就可以在你google drive的data 目錄下看 到以下檔案

• 中斷執行階段後就可以執行下一步驟

📥 雲端硬碟	Q、在雲端硬碟中搜尋	₽
+ 新增	我的雲端硬碟 > data +	
▲ 首頁	(類型 ▼) 使用者 ▼ 上次修改日期 ▼	
▶	☆ 鍵盤快速鍵更新 Google 雲端硬碟將在 2024年8月1日 星期四更新鍵	盤快速鍵,讓你透過首字母導覽功能
・「自己の問題」		
3、 與我共用	名稱	擁有者
③ 近期存取	JPEGImages	8 我
☆ 已加星號	labels	❷ 我
① 垃圾内容	🖹 all_train.txt	₿ 我
■ 垃圾桶		0.0
△ 儲存空間	i val.txt	10 技
- 目前使用量: 616.3 MB (儲	E train.txt	9 我
行工间配額: 15 GB)	E test.txt	④ 我
-V13~2 mm (11B)	E trainval.txt	④ 我





打開Train_YOLO_Object_Detection_With_Colab

÷	→ C == colab.research.go	ogle.com/drive/1c2s2VC3F7Zjn80jF3MJyK	l6ziKxYRSIX?authuser=2	٤ 🗹				
11 應月		ACUT 🗋 LINE 🗋 論文 🗋 交通 🔇 A	AFM-原子力顯微鏡 🧿 Speedtest by Ookl					
co	▲ 在Colab訓練YOLOv4-1 檔案編輯檢視畫面插入報	tiny 模型.ipynb ☆ 約時 日月 説明 <u>最近於 4月25日 編</u> 輯	■ 留言	▲ 共用				
i = 1	目録 C ×	+ 程式碼 + 文字	✓ T4 RAM — •	• 🛛 😔 C				
۹	Running a YOLOv4 Object Detector with	Running a YOLOv4 Object	連線至代營的執行階段: T4 變更執行階段類型	loud				
{x}	Darknet in the Cloud! (GPU ENABLED)	ENABLED)	連線至自訂的 GCE VM 連線至本機執行階段					
	Step 1: Enabling GPU within your notebook	This tutorial will help you build YOLC object detections in milliseconds!	查看資源 67-00-17 (ABMC)	that you				
	Step 3: Download pre-	Step 1: Enabling GPU \-	管理工作階段 中斷連線並刪除執行階段					
	trained YOLOv4 weights		顯示執行過的程式碼歷史記錄					
	Step 4: Define Helper Functions	will be able to process detections over	er 100 times faster than CPU.					
	Step 5: Run Your Detections with Darknet and YOLOv4!	Steps:						
	换為自己的AI模型for HUB 8735	I) Click Edit at top left of your	notebook					
<>	需額外操作	File Edit View Insert R	untime Tools Help <u>All changes s</u>					
	測試模型	≔ Files	× + Code + Text					
2	其他資料	ii) Click Notebook Settings wi	thin dropdown					

變更執行階段類型
執行階段類型
Python 3 👻
硬體加速器 ⑦
O CPU O T4 GPU A100 GPU O L4 GPU
V100 GPU (deprecated) O TPU (deprecated)
O TPU v2
想要使用付費 GPU 嗎? 購買額外運算單元
取消儲存





其他資料

補充

<>

執行完成後就可以看到這張圖片

• 現在·將可以開始訓練自己的AI模型了

• 首先執行下列3行程式

CO A Colab訓練YOLOv4-tiny 視型.ipynb 🕱
【】
田 ★ + 程式碼 + 文字 Assertionerror: Asertioneroro: Assertionerror: Assertionerror:
Next steps: Explain error
^{{𝔅} } → □ darknet
☞ ト 🖿 3rdparty · 換為自己的AI模型for HUB 8735
▶ ■ backup
▶ ■ build my_yolov4-tiny
► cfg
I cmake I cmake I drive. mount (' <u>/content/gdrive</u> ')
▶ data
▶ ▶ include !cp -r <u>/content/gdrive/MyDrive/data</u> /* <u>/content/darknet</u>
▶ ■ obj ‱kdir /content/gdrive/MvDrive/data/results
▶ results
▶ scripts
· · · · · · · · · · · · · · · · · · ·
CMakeLists 於此手動上傳,編輯過之: my_ai.data, my_ai.names, my_yolov4-tiny.cfg











將3個檔案上傳自darknet

10180



• 這樣就可以開始訓練你的模型了

7

C		住Colaball線YOLOV4- 編輯 檢視畫面 插入	-tiny <mark>侯坐.ipynb 🛛 🛪</mark> 執行階段 工具 說明	\sim						E f
	+程:	弌碼 + 文 字			1.					■ RAM ■ T4 磁碟 ■
٩	~	需額外操作			7					
{ <i>x</i> }	於山	比手動上傳,編輯過之: m	y_ai.data, my_ai.names,	my_yolov4-tiny.cfg		5				
©~	 [15]	%cd <u>/content/darknet</u> %1s			MS					
		<pre>/content/darknet 3rdparty/ all_train.txt backup/ build/ build.psl* cfg/ cmake/ C(MakeLists.txt darknet* DarknetConfig.cmake.in darknet.py</pre>	<pre>darknet_video.py data/ docker-compose.yml Dockerfile.cpu Dockerfile.gpu image/ image_yolov3.sh* include/ json_mjpeg_streams.sl labels/ LICENSE</pre>	Makefile my_ai.data my_yolov4-tiny.cfg net_cam_v3.sh* net_cam_v4.sh* obj/ package.xml predictions.jpg n* README.md results/ scripts/	<pre>src/ test.txt train.txt trainval.txt vcpkg.json vcpkg.json.opencv23 video_yolov3.sh* video_yolov4.sh* yolov4-tiny.conv.29 yolov4-tiny.weights</pre>		, 90	1 *	S E	↑ ↓
	0	▶!./darknet detector t	train my_ai.data my_yol	.ov4-tiny.cfg yolov4-	tiny.conv.29 -dont_shc	Ŵ		- Sh		
		CUDA-version: 12020 (1 CUDNN_HALF=1 OpenCV version: 4.5.4 my_yolov4-tiny 0 : compute_capability net.optimized_memory = min betch = 64	2020), cuDNN: 8.9.6, CUD r = 750, cudmn_half = 1, 0 = 64 time steps = 1 th	NN_HALF=1, GPU count: GPU: Tesla T4				-24	1	Ļ,
< >		layer filters siz 0 Create CUDA-stream Create cudnn-bandle 0	re/strd(di1) input n = 0	output					7414	
		conv 32 3 x 3 1 conv 64 2 conv 64 3 route 2	3/2 416 x 416 x 3 - 3 x 3/2 208 x 208 x 3 x 3/1 104 x 104 x	> 208 x 208 x 32 0. 32 -> 104 x 104 x 64 -> 104 x 104 x 1/2 -> 104 x 104 x	075 BF 64 0.399 BF 64 0.797 BF 32				N	





- 若想使用Yolov7模型可下載以下檔案
- Yolov7-tiny
 - cfg: https://raw.githubusercontent.com/AlexeyAB/darknet/master/cfg/yolov7-tiny.cfg
 - weights for fine-tuning: <u>https://github.com/AlexeyAB/darknet/releases/download/yolov4/yolov7-tiny.conv.87</u>
- 修改my_yolov7-tiny.cfg後將2個檔案上傳自darknet 目錄下
- 修改





訓練完之後就可以在data/result看到你的權重檔

我的	雲端硬碟 > data > results -			✓ = :::	()
類型	▼ 使用者 ▼ 上次修改日期 ▼				
名稱	↑	擁有者	上次修改時間 🔻	檔案大小	:
	my_yolov4-tiny_last.weights	8 我	下午5:35	22.5 MB	÷



AI模型轉換

把你的檔案下載下來並和my_yolov4-tiny.cfg壓縮成 zip

我的雲端硬碟 > data >	results -		4	1	× +			_ 0
× 已選取1個 :				$\leftarrow \rightarrow \uparrow$	С□≻н	JB 8735 > 1	搜尋 1	
5稱 个		上次修 ▼		(†) 新增 🗸 👗	0 🗈 🗳	e û	1↓排序 ~ •••	□ 詳細
my_yolov4-tiny_last.weights		下午5:39		☆常用	名稱		修改日期	類型
	◆ 選擇開啟工具			🛃 圖庫	📄 my_ai.data		2024/5/6 下午 04:00	DATA 檔案
				📥 楊仕鵬 - 明	📄 my_ai.names	Л	2024/5/6 下午 04:01	NAMES 檔案
	▶ 下載			GEAD	📄 my_yolov4-tiny.cfg		2024/5/6 下午 02:32	Configuratio
	/ 重新命名				my volov4-tiny last	(1)aveightsfiguration	_动 建建造/6 下午 05:42	WEIGHTS 檔
	🗋 建立副本	Ctrl+C Ctrl+V				大小: 3.14 KB	下午 02-22	
	2+ #用	•		🛓 下載 🔷				
				📄 文件 🔺		7	2	
				🔀 🗐 片 🖌	5. </td <td></td> <td>y</td> <td></td>		y	
		•		● 音樂 ●	'' N (/ / /		. .	
	■ 移至垃圾桶						7/	
				22 影片 🖈				
				🖿 螢幕擷取畫				

🚞 example

AI模型轉換

● 至<u>https://www.amebaiot.com/zh/amebapro2-ai-convert-model/</u>進行線上AI 模型轉換

CNN-RGB or CNN-GRAY)		
YOLO-TINY	選擇YOLO-TINY	~
Quantize Type (required, UII	NT8 or INT16)	
INT16	選擇一種量化類型·UINT8或INT16都可以	~
Upload zip file including a cf file contained the ".cfg" and "	g file and a weights file(required, please upload the folder or comp ".weights" files, all named in English, limit:35MB)	iressed
選擇檔案 my_yolov4-t	iny.zip 選擇zip檔上傳至此·檔案大小限制3	5MB
Upload one jpg file (required	J, limit:1MB)	
選擇檔案 0 (21).jpg	選擇任何一張圖片來驗證	
Upload jpg files (option, limi	t:1MB)	
選擇檔案 沒有選擇檔案		
選擇檔案 沒有選擇檔案		



以Arduino IDE將模型燒入晶片

- 1.等待自動回覆至登記註冊e-mail 信箱裡,並將nb檔案下載。
- 2.開啟資料夾至:

C:\Users\<使用者名稱>\AppData\Local\Arduino15\packages\id easHatch\hardware\AmebaPro2\4.0.10-Release\variants\comm on_nn_models

● 替換yolov4_tiny.nb 檔案。





以Arduino IDE將模型燒入晶片

開啟程式範例>AmebaNN>ObjectDetectionCallback





以Arduino IDE將模型燒入晶片







硬體準備(電燈)





程式說明

#加入 servo函式庫

#include <AmebaServo.h>
AmebaServo myservo;

34

#在setup裡定義腳位

124 pinMode(7, OUTPUT);
125 pinMode(9, OUTPUT);
126 myservo.attach(8);

#設定功能

3 × 4	<pre>if (itemList[obj_type].index == 0) { digitalWrite(7, LOW); }</pre>
5 ~ 7 3	<pre>if (itemList[obj_type].index == 1) { digitalWrite(7, HIGH); }</pre>
2 × 0 2 3 4 5	<pre>if (itemList[obj_type].index == 2) { for (pos = 0; pos <= 100; pos += 1) { // tell servo to go to position in variable 'pos' myservo.write(pos); delay(15); } delay(1000); }</pre>
5 × 7 × 3 9 9	<pre>if (itemList[obj_type].index == 3) { for (pos = 100; pos >= 0; pos -= 1) { // tell servo to go to position in variable 'pos' myservo.write(pos); delay(15); } delay(1000); }</pre>
3 V 4	<pre>if (itemList[obj_type].index == 4) { digitalWrite(9, HIGH); }</pre>
5 ~ 7 3	<pre>if (itemList[obj_type].index == 5) { digitalWrite(9, LOW); } </pre>

成果展示

成果如影片所示 ● 手勢0:關閉電燈 • 手勢1:開啟電燈 • 手勢2:開啟窗戶 手勢3:關閉窗戶 • 手勢4:開啟電扇 ● 手勢5:關閉電扇



Thanks.

Every failure is a step to success.