人工智慧

口罩辨識系統製作

吳智鴻 教授

國立臺中教育大學 數位內容科技學系

資料來源

參考網站：

https://github.com/chandrikadeb7/Face-Mask-Detection

系統架構

使用到的架構

⚠ **TechStack/framework used**

- OpenCV
- Caffe-based face detector
- Keras
- TensorFlow
- MobileNetV2

MobileNet v1

谷歌 2017 年推出了 MobileNetV1，它是一種為移動設備設計的通用計算機視覺神經網絡，因此它也能支持圖像分類和檢測等。一般在個人移動設備上運行深度網絡能提升用戶體驗、提高訪問的靈活性，以及在安全、隱私和能耗上獲得額外的優勢。

此外，隨著新應用的出現，用戶可以與真實世界進行實時交互，因此我們對更高效的神經網絡有著很大的需求。

原文網址：https://kknews.cc/tech/nekyn68.html

MobileNet V2

下一代移動視覺應用 MobileNetV2 已經發布。

MobileNetV2 在 MobileNetV1 的基礎上獲得了顯著的提升，並推動了移動視覺識別技術的有效發展，包括分類、目標檢測和語義分割。

MobileNetV2 作為 TensorFlow-Slim 圖像分類庫的一部分而推出，讀者也可以在 Colaboratory 中立即探索 MobileNetV2。

原文網址：https://kknews.cc/tech/nekyn68.html

MobileNetv2架構

可以直觀理解為，瓶頸層對模型的中間輸入與輸出進行編碼，而內層封裝了模型從像素等低級概念到圖像類別等高級概念的轉換能力。最後，與傳統的殘差連接一樣，捷徑能快速訓練並獲得更優精確度。

原文網址：
https://kknews.cc/tech/nekyn68.html
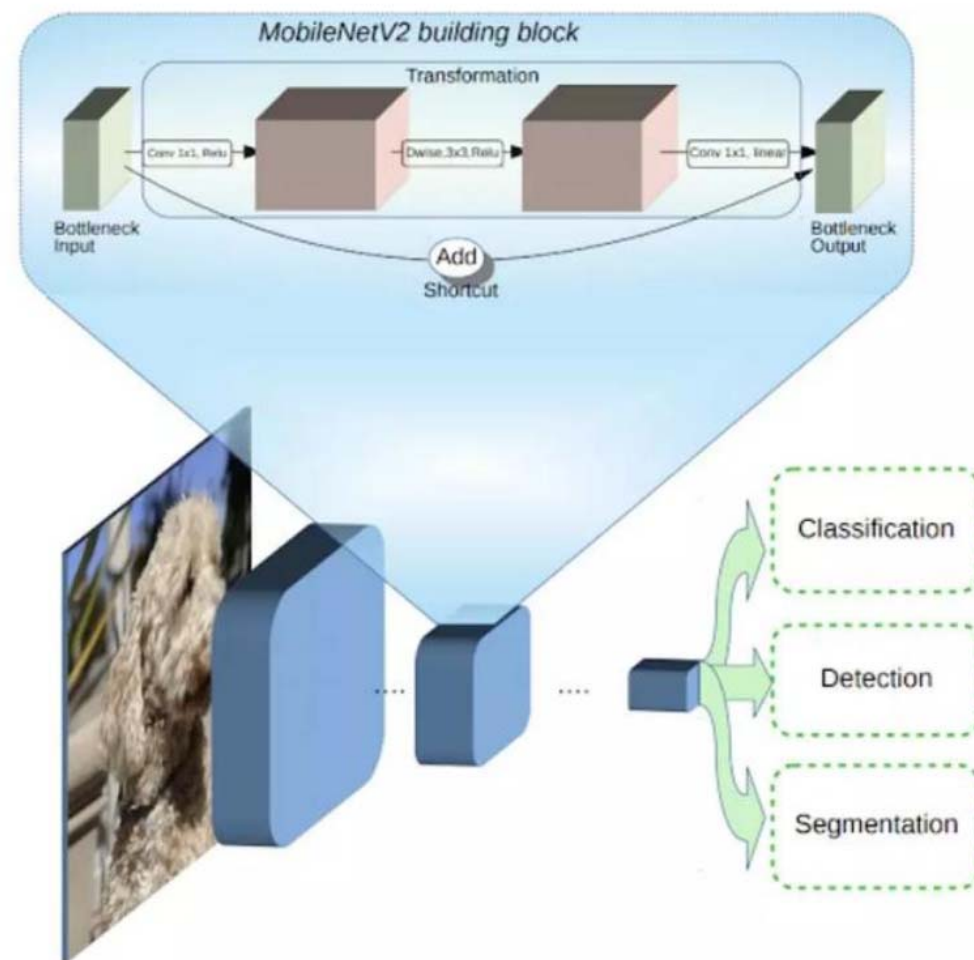
MobileNetV2 的架構概覽，藍色塊如上所示為複合卷積構建塊。

MobileNet v2的優點

V2 與第一代的 MobileNet 相比有什麼區別？

總體而言，MobileNetV2 模型在整體延遲範圍內上實現相同的準確度要更快。

特別是，目前新模型減少了兩倍 operations 的數量，且只需要原來 70% 的參數，在 Google Pixel 手機上的測試表明 V2 要比 MobileNetV1 快 30% 到 40%，同時還能實現更高的準確度。

原文網址：https://kknews.cc/tech/nekyn68.html

⭐ Features

Our face mask detector didn't use any morphed masked images dataset. The model is accurate, and since we used the MobileNetV2 architecture, it's also computationally efficient and thus making it easier to deploy the model to embedded systems (Raspberry Pi, Google Coral, etc.).

This system can therefore be used in real-time applications which require face-mask detection for safety purposes due to the outbreak of Covid-19. This project can be integrated with embedded systems for application in airports, railway stations, offices, schools, and public places to ensure that public safety guidelines are followed.

採用MobileNetV2架構。 可部屬在 Raspberry Pi, Google Coral等嵌入式系統上

資料集來源

📁 **Dataset**

The dataset used can be downloaded here - Click to Download

This dataset consists of **3835 images** belonging to two classes:

- **with_mask: 1916 images**
- **without_mask: 1919 images**

The images used were real images of faces wearing masks. The images were collected from the following sources:

- **Bing Search API** (See Python script)
- **Kaggle datasets**
- **RMFD dataset** (See here)

🚀 Installation

1. Clone the repo

```
$ git clone https://github.com/chandrikadeb7/Face-Mask-Detection.git
```

2. Change your directory to the cloned repo and create a Python virtual environment named 'test'

```
$ mkvirtualenv test
```

3. Now, run the following command in your Terminal/Command Prompt to install the libraries required

```
$ pip3 install -r requirements.txt
```

安裝方式

1. git hub下載原始檔

   https://github.com/chandrikadeb7/Face-Mask-Detection

   解開後放在 c:\ai\face-mask-detection-master目錄下

2. 打開anaconda prompt

   python.exe -m pip install —upgrade pip   (更新 pip指令)
   conda create -n mask python=3.8   (創建一個虛擬環境 mask)

   conda activate mask   (啟動虛擬環境)

   pip3 install -r requirements.txt   (安裝本project所需套件)

💡 Working

1. Open terminal. Go into the cloned project directory and type the following command:

```
$ python3 train_mask_detector.py --dataset dataset
```

2. To detect face masks in an image type the following command:

```
$ python3 detect_mask_image.py --image images/pic1.jpeg
```

3. To detect face masks in real-time video streams type the following command:

```
$ python3 detect_mask_video.py
```

🔑 Results

Our model gave 93% accuracy for Face Mask Detection after training via `tensorflow-gpu==2.0.0`

```
[INFO] evaluating network...
               precision    recall   f1-score    support

    with_mask       0.99      0.86       0.92        383
 without_mask       0.88      0.99       0.93        384

     accuracy                            0.93        767
    macro avg       0.93      0.93       0.93        767
 weighted avg       0.93      0.93       0.93        767

[INFO] saving mask detector model...
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```
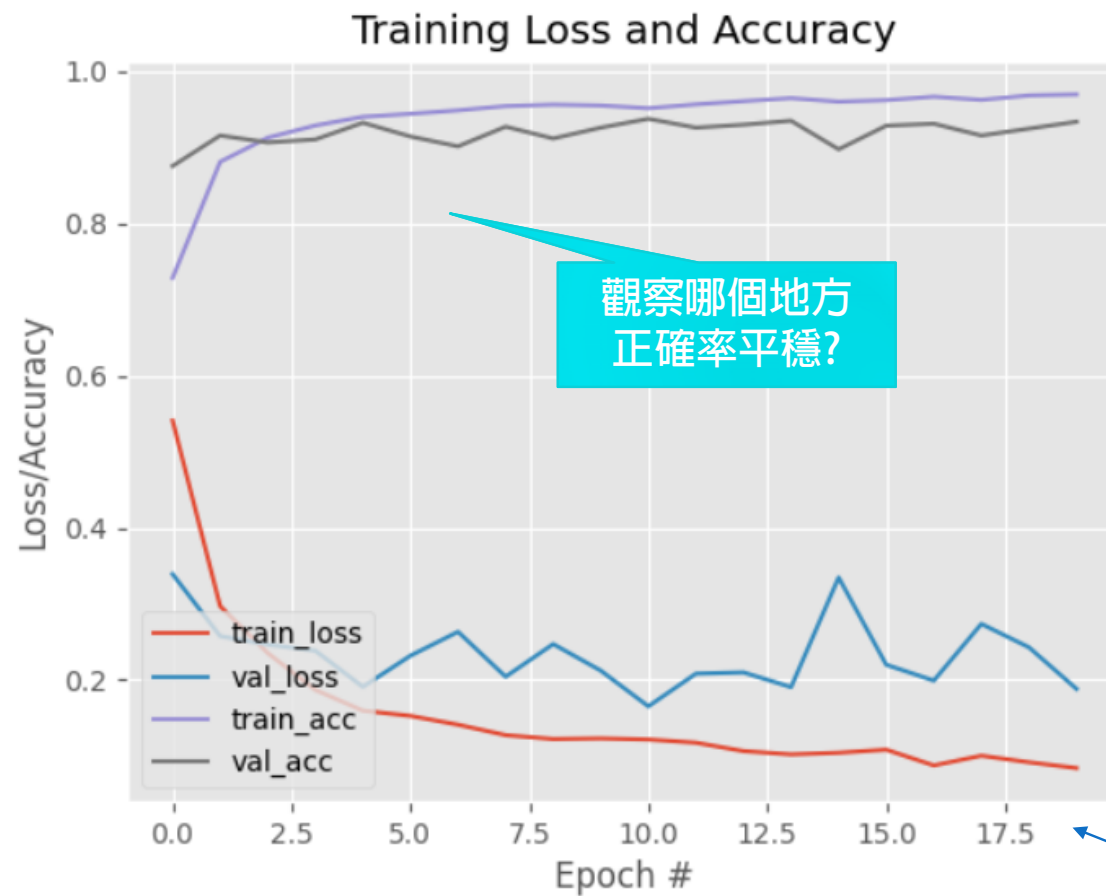
We got the following accuracy/loss training curve plot

Training Loss and Accuracy

觀察哪個地方
正確率平穩？

觀察哪個地方
誤差最低

Exercise#1 完成一張口罩圖片的辨識

指令:

準備一張自己的口罩照片，放在images目錄下

python detect_mask_image.py ─image images/xxx.jpeg

將結果截圖上傳FB社團

準備一張照片
images底下

命令列執行：
python detect_mask_image.py –image images/pic4.png

辨識結果

Exercise#2

修改訓練的次數，從20次-> 5次 觀察訓練成效acc & loss

修改 train_mask_detector.py

```
38    # initialize the initial learning rate, number of epochs to train for,
39    # and batch size
40    INIT_LR = 1e-4
41    EPOCHS = 20
42    BS = 32
```

觀察哪個地方
要修改?

python train_mask_detector.py --dataset dataset

Exercise#3 增加自己的訓練圖片

增加幾張自己的口罩圖片。放在dataset底下

重新訓練模型

Exercise#3 增加自己的訓練圖片 (做法)

增加幾張自己的口罩圖片。放在dataset底下

📁 with_mask	2020/12/16 下午 ...	檔案資料夾
📁 without_mask	2020/12/16 下午 ...	檔案資料夾

有口罩&無口罩的圖片

重新訓練模型

Python train_mask_detector.py —dataset dataset

Exercise#4 啟用webcam自動辨識口罩並修改顯示內容

要求：

啟動webcam辨識口罩程式

修改顯示內容，

修改 Mask & No mask的顯示文字

修改 Mask & No mask的顯示外框顏色

顯示後結果截圖上傳FB社團

Exercise#4 (解答)
啟用webcam自動辨識口罩並修改顯示內容

要求：

啟動webcam辨識口罩程式

python detect_mask_video.py

修改顯示內容，顯示後結果截圖上傳

```
118   for (box, pred) in zip(locs, preds):
119       # unpack the bounding box and predictions
120       (startX, startY, endX, endY) = box
121       (mask, withoutMask) = pred
122
123       # determine the class label and color we'll use to draw
124       # the bounding box and text
125       label = "Mask" if mask > withoutMask else "No Mask"
126       color = (0, 255, 0) if label == "Mask" else (0, 0, 255)
127
128       # include the probability in the label
129       label = "{}: {:.2f}%".format(label, max(mask, withoutMask) * 100)
130
131       # display the label and bounding box rectangle on the output
132       # frame
133       cv2.putText(frame, label, (startX, startY - 10),
134           cv2.FONT_HERSHEY_SIMPLEX, 0.45, color, 2)
135       cv2.rectangle(frame, (startX, startY), (endX, endY), color, 2)
```

資料來源

1. https://github.com/chandrikadeb7/Face-Mask-Detection

2. Mobilenet v2介紹

https://medium.com/ai-academy-taiwan/efficient-cnn-%E4%BB%8B%E7%B4%B9-%E4%BA%8C-mobilenetv2-7809721f0bc8

3.谷歌發布MobileNetV2：可做語義分割下一代移動端計算機視覺架構

原文網址：https://kknews.cc/tech/nekyn68.html

Congratulations.

You can design your real-time Mask Detection now！