

機器學期其中專題

天牛 VS 獨角仙

109321001 吳長恩

Using tensorflow 分辨天牛和
獨角仙?

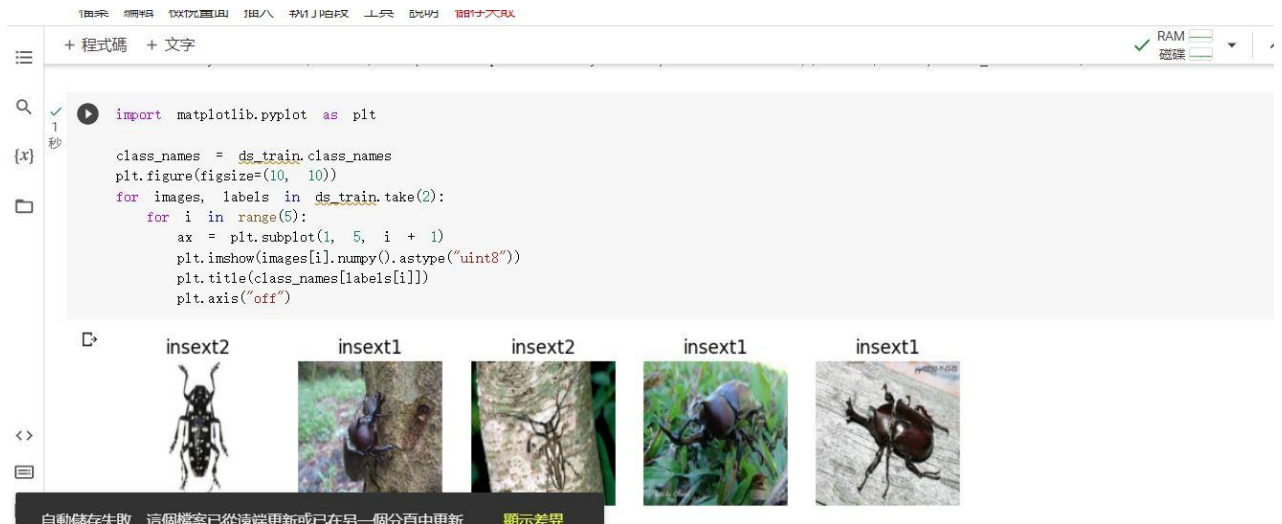
第一步匯入資料

```
tf.keras.utils.image_dataset_from_directory
```

這 tensorflow framework 好用及多功能於一身，自動幫你把不同大小的圖形都設定好(我這是設(240*240)還自動幫你洗牌拆成 validation 和 training data

第二步 plot data 看看是不是我所預期

以下是前五個 training data



我突然發現第一步應該是找圖片

我一開始想說會不會需要同樣大小的圖片才行，會到第一頁所提到的函數其實不用特別處理但是 **resize** 之後樣子的確怪怪的

總之最後我各找了 25 張，所以總共 50 張圖片

第三步建立模型

Mlp 和 cnn 我都試

mlp:accuracy :40%

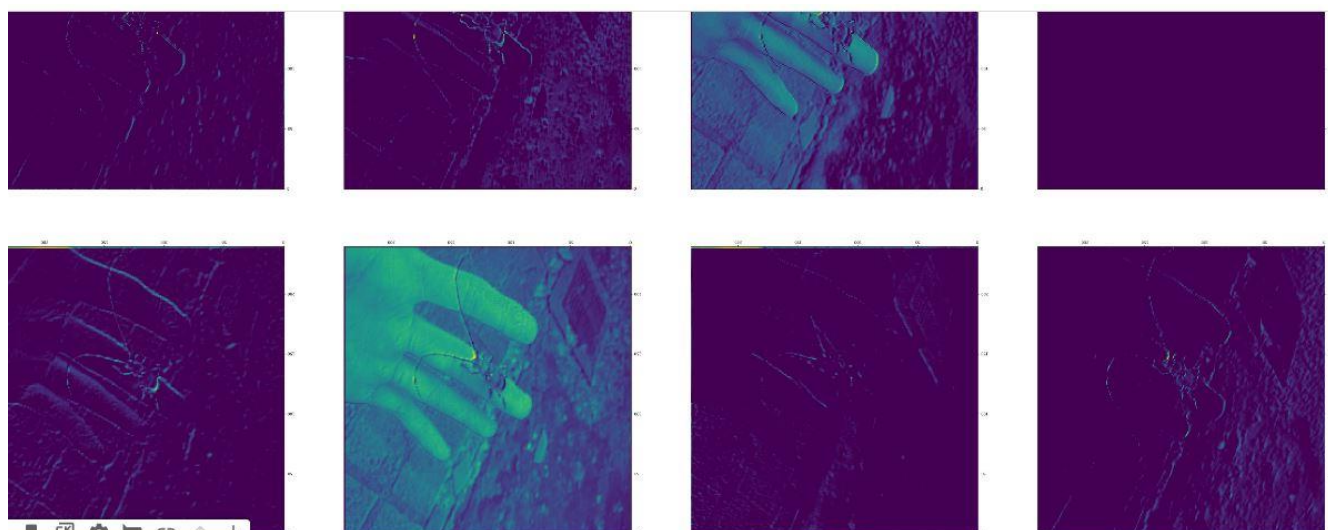
Cnn(using only 1 layer)

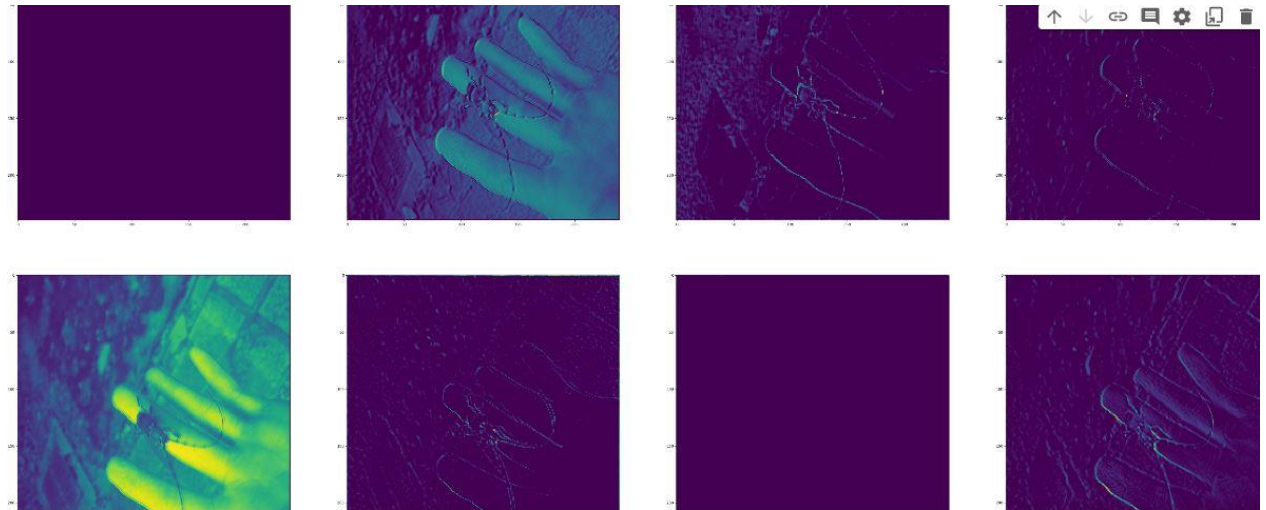
:accuracy:100%

中間遇到的小問題:

dense 太少(64 個)我還以為
我做錯了 我調到 512 就成功
了

Cnn filter plot(看看它輪廓
(contour)處理如何)





原圖

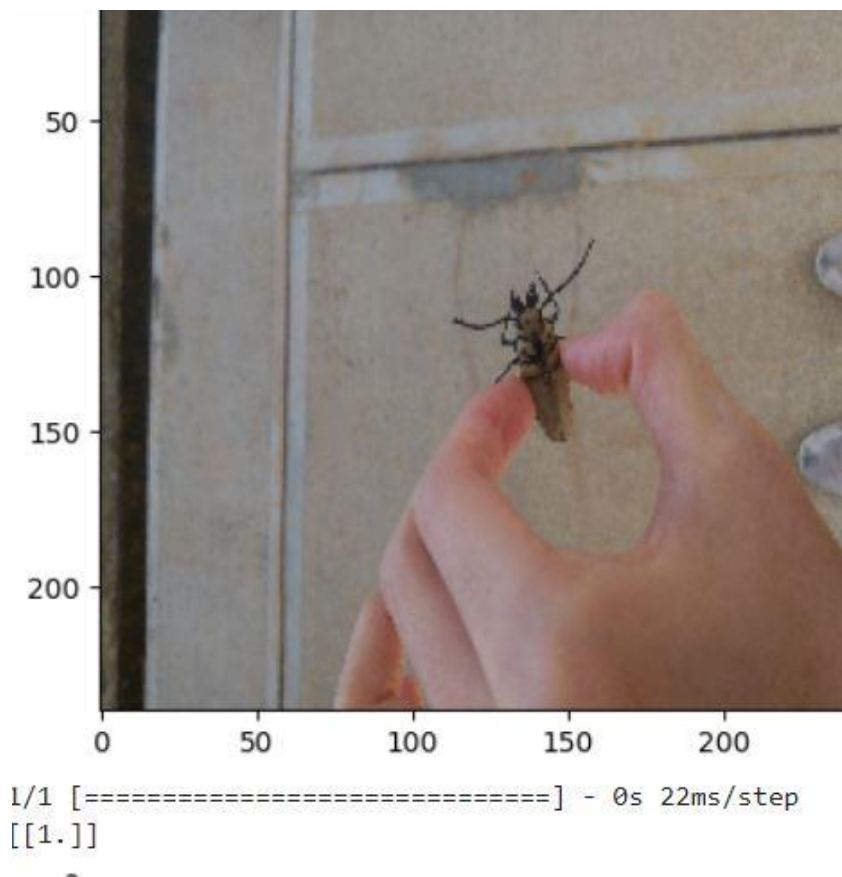


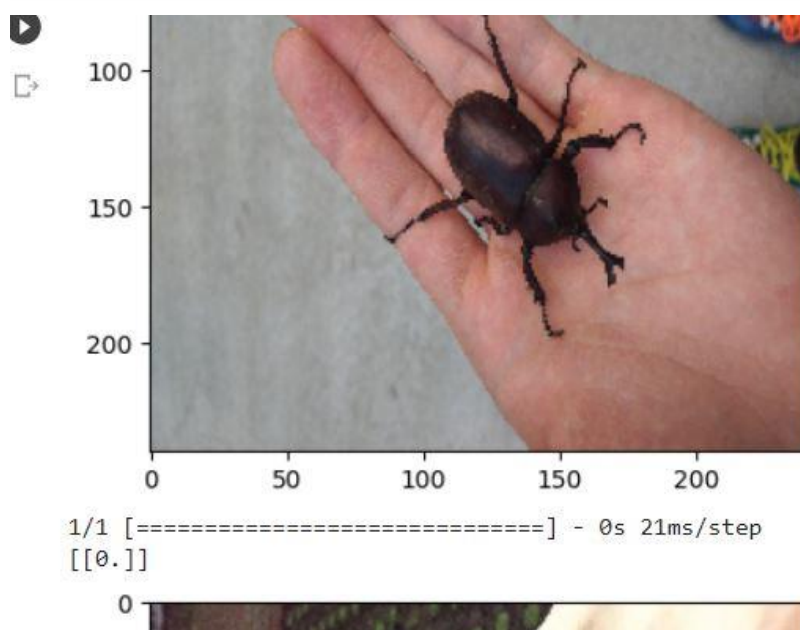
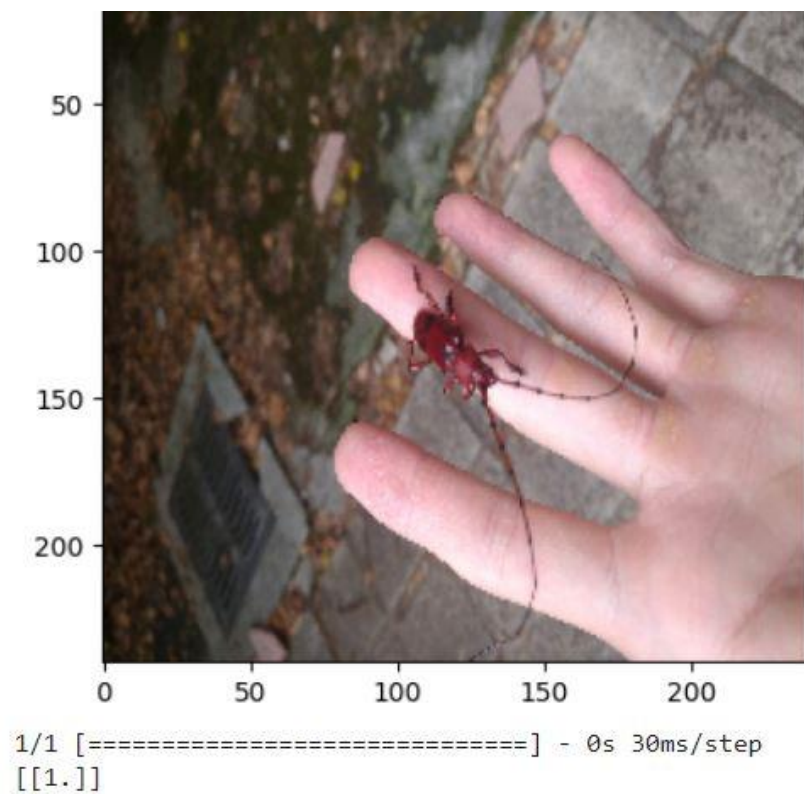
結果

```
Epoch 1/10
9/9 [=====] - 1s 28ms/step - loss: 35294.7773 - acc: 0.5333
Epoch 2/10
9/9 [=====] - 0s 25ms/step - loss: 3936.3752 - acc: 0.6667
Epoch 3/10
9/9 [=====] - 0s 24ms/step - loss: 4216.8057 - acc: 0.4444
Epoch 4/10
9/9 [=====] - 0s 25ms/step - loss: 1578.9866 - acc: 0.7778
Epoch 5/10
9/9 [=====] - 0s 24ms/step - loss: 493.6948 - acc: 0.8667
Epoch 6/10
9/9 [=====] - 0s 24ms/step - loss: 2676.9360 - acc: 0.5333
Epoch 7/10
9/9 [=====] - 0s 25ms/step - loss: 872.3824 - acc: 0.7111
Epoch 8/10
9/9 [=====] - 0s 25ms/step - loss: 167.0785 - acc: 0.9111
Epoch 9/10
9/9 [=====] - 0s 26ms/step - loss: 0.0000e+00 - acc: 1.0000
Epoch 10/10
9/9 [=====] - 0s 26ms/step - loss: 0.0000e+00 - acc: 1.0000
<keras.callbacks.History at 0x7f34564aff70>
```

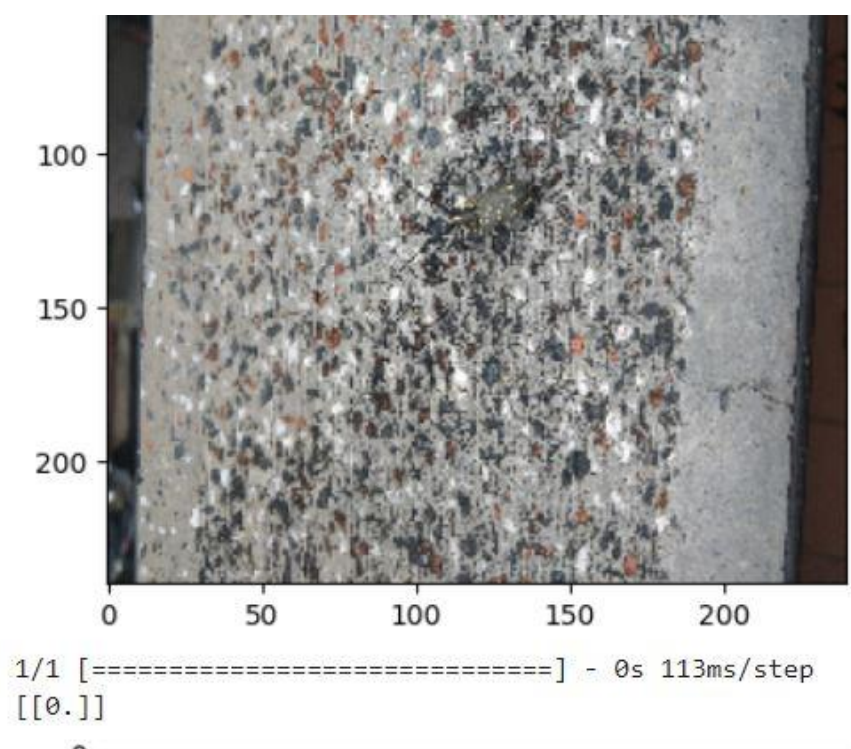
最後 prediction

圖片不是 training set(網路上找的)是自己在學校抓的





這張就錯了



結論

電腦還是很笨，我給我表弟看個六張
他就會分了，我給她看五十張它還可
以分錯

Github

https://github.com/109321001/1112ncnu_machine_learning

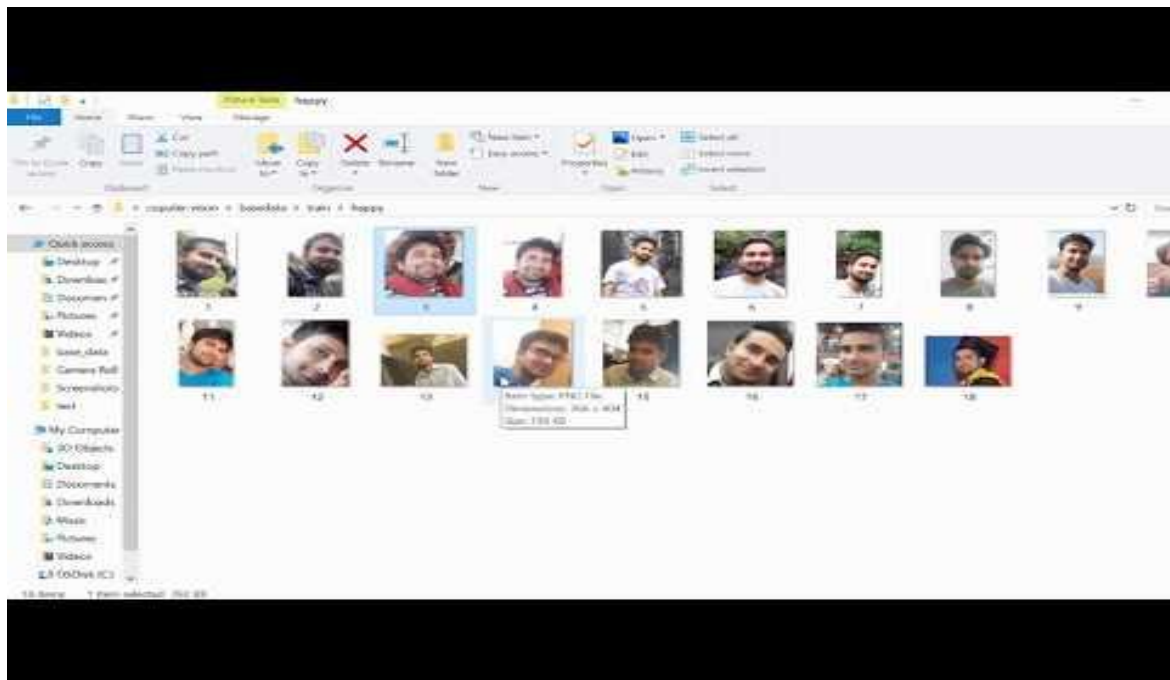
參考資料

[152 - How to visualize convolutional filter outputs in your deep learning model?](#)



How to visualize convolutional layer responses in your deep learning model?

[Train Neural Network by loading your images | TensorFlow, CNN, Keras tutorial](#)



TensorFlow Tutorial 05 - Convolutional Neural Network (CNN)



