

目的：辨別猜拳時出的是剪刀、石頭，還是布

資料來源：Kaggle 上抓取

<https://www.kaggle.com/drgfreeman/rockpaperscissors>

利用遷移學習使用三種模型訓練

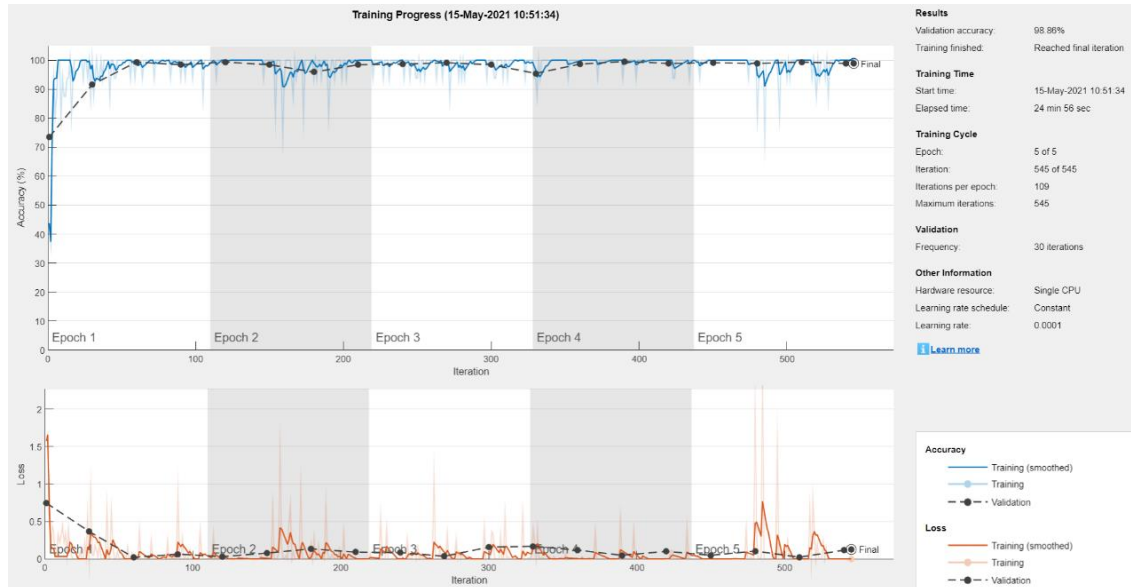
1. AlexNet

於課堂作法一樣，更改最後三個層

將 input 層的 inputsize 改為[320,240,3]

批次大小 16，迭代 5，學習率 0.0001

```
%Set Training Options 設定網路訓練參數
Checkpoint=pwd;
options = trainingOptions('adam',...
    'MiniBatchSize',16,...
    'MaxEpochs',5,...
    'InitialLearnRate',1e-4,...
    'ValidationData',augimdsTest,...
    'ValidationFrequency',30,...
    'ValidationPatience',Inf,...
    'Verbose',1,...
    'ExecutionEnvironment','auto',...
    'Plots','training-progress');
```



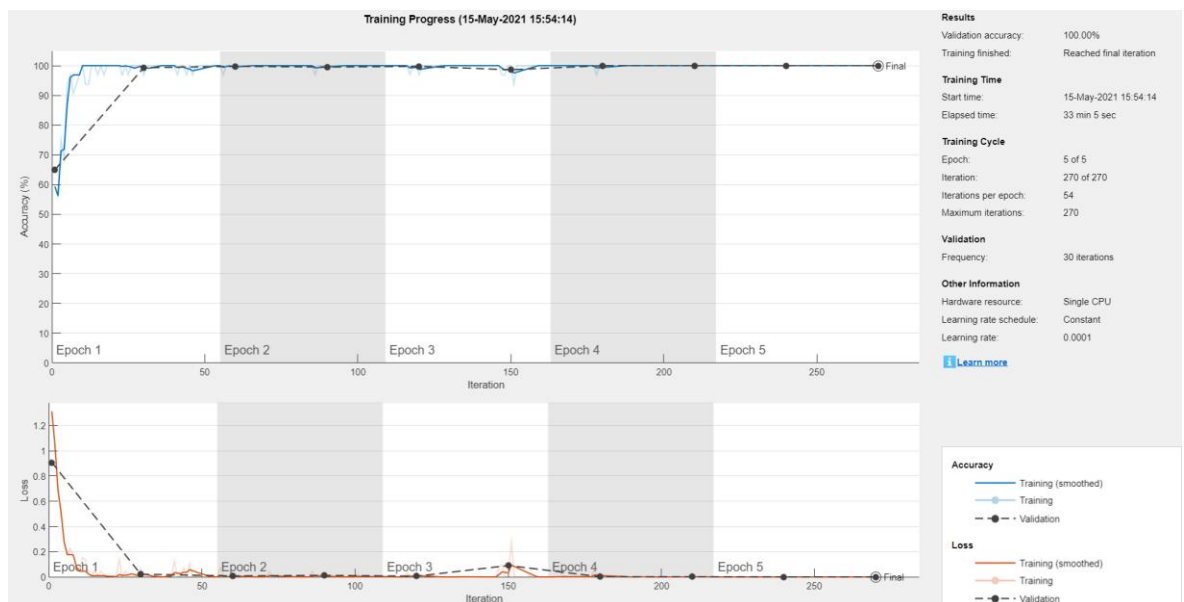
由上圖可以發現，迭代 5 次而已，其實訓練就已經收斂，再加上出來的準確度以及 loss 是不錯的，因此就沒有想把迭代次數增加的想法

Confusion Matrix				
Output Class	paper	rock	scissors	
	<div>138 31.6%</div>	<div>0 0.0%</div>	<div>1 0.2%</div>	<div>99.3% 0.7%</div>
	<div>2 0.5%</div>	<div>145 33.2%</div>	<div>0 0.0%</div>	<div>98.6% 1.4%</div>
	<div>2 0.5%</div>	<div>0 0.0%</div>	<div>149 34.1%</div>	<div>98.7% 1.3%</div>
	<div>97.2% 2.8%</div>	<div>100% 0.0%</div>	<div>99.3% 0.7%</div>	<div>98.9% 1.1%</div>
	paper	rock	scissors	
Target Class				

2. GoogLeNet

先用與做 AlexNet 時所更改的方式去訓練，參數部分不同的是批次大小為 32

```
%Set Training Options 設定網路訓練參數
Checkpoint=pwd;
options = trainingOptions('adam',...
    'MiniBatchSize',32,...
    'MaxEpochs',5,...
    'InitialLearnRate',1e-4,...
    'ValidationData',augimdsTest,...
    'ValidationFrequency',30,...
    'ValidationPatience',Inf,...
    'Verbose',1,...
    'ExecutionEnvironment','auto',...
    'Plots','training-progress');
```



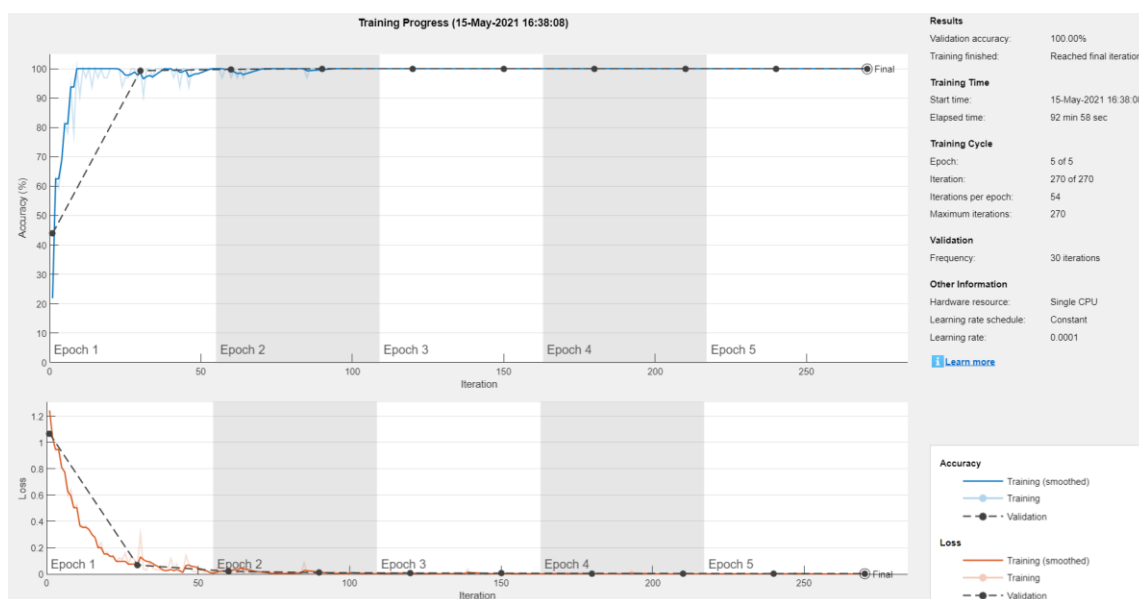
其實可以發現效果是比 AlexNet 好，且準確度已經 100%，因此也不做任何更改

Confusion Matrix				
Output Class	paper	rock	scissors	
	<div>142 32.5%</div>	<div>0 0.0%</div>	<div>0 0.0%</div>	<div>100% 0.0%</div>
	<div>0 0.0%</div>	<div>145 33.2%</div>	<div>0 0.0%</div>	<div>100% 0.0%</div>
	<div>0 0.0%</div>	<div>0 0.0%</div>	<div>150 34.3%</div>	<div>100% 0.0%</div>
	<div>100% 0.0%</div>	<div>100% 0.0%</div>	<div>100% 0.0%</div>	<div>100% 0.0%</div>
Target Class				
	paper	rock	scissors	

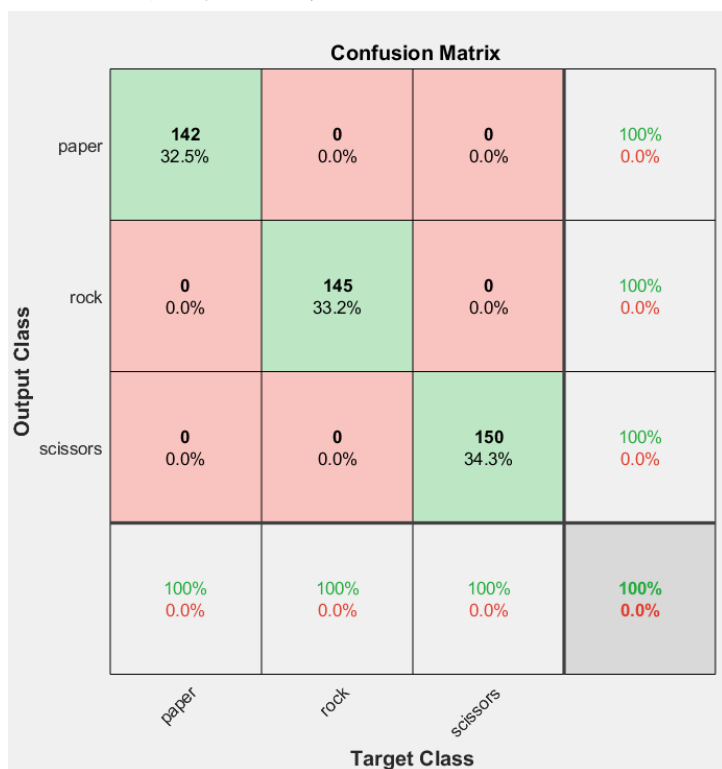
3. EfficientNet-b0

先以訓練 GoogLeNet 模型時的參數設定去訓練

```
%Set Training Options 設定網路訓練參數
Checkpoint=pwd;
options = trainingOptions('adam',...
    'MiniBatchSize',32,...
    'MaxEpochs',5,...
    'InitialLearnRate',1e-4,...
    'ValidationData',augimdsTest,...
    'ValidationFrequency',30,...
    'ValidationPatience',Inf,...
    'Verbose',1,...
    'ExecutionEnvironment','auto',...
    'Plots','training-progress');
```



可看出效果一樣好，準確度高達 100%



結論：

因為利用三個模型進行遷移學習的準確度幾乎是 100%，這個結果是好到自己會有點懷疑，畢竟準確度 100% 不是很常出現、很容易做得出來的，因此去 Kaggle 上看看，其他人做出來的情況也是這麼高，所以造成這個情形發生推測是資料組蠻乾淨，幾乎沒有雜訊，以及此分類難度不高，才出現幾乎是 100% 的現象。