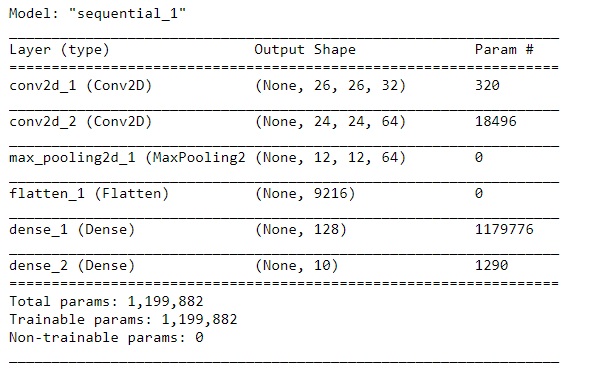
**機器學習系統設計實務與應用**

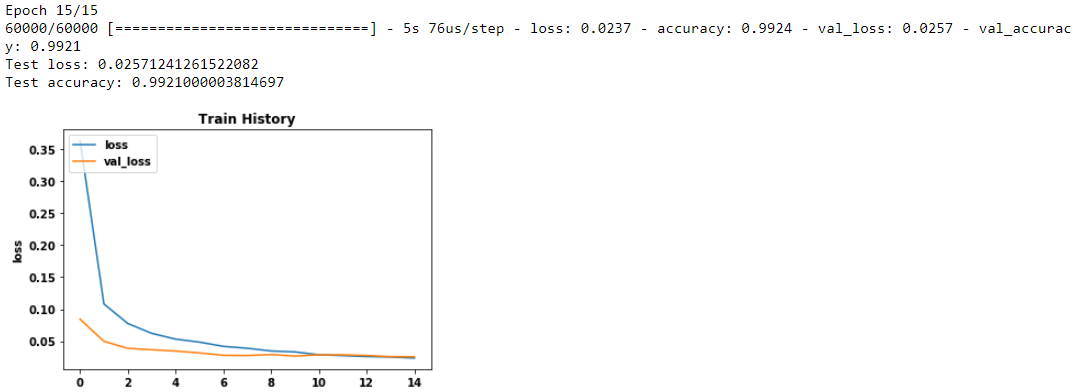
HW3 B063012054 林祐安

於前一次作業問題與討論提到：

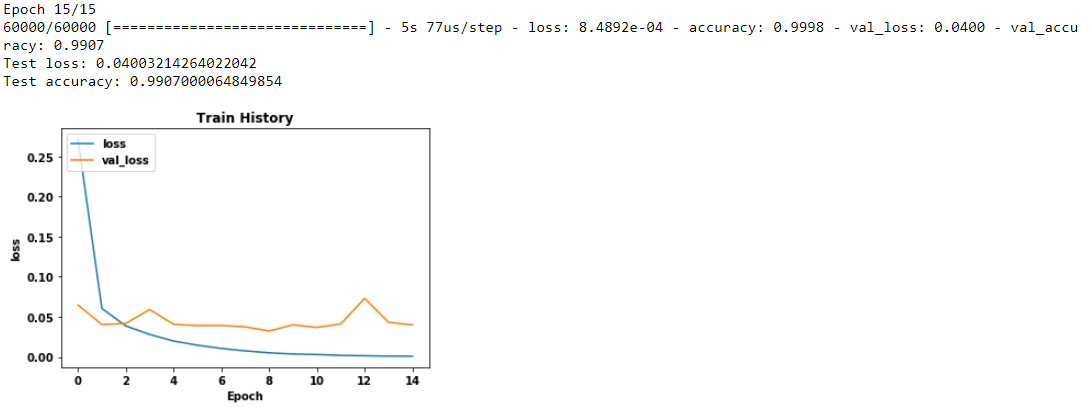
原架構Dropout->Dense->Dropout->Dense，因圖片單純、架構簡單，拿掉兩個Dropout是不是沒有太大影響？



(刪除兩層Dropout模型架構)



(尚未修改前的模型)



(刪除兩層Dropout的模型)

比較兩者趨勢圖，很明顯看到”刪除兩層Dropout的模型”於valid loss方面震盪，原以為loss無法下降，但仔細觀察其值為0.05左右，與”尚未修改前的模型”loss相差約0.015，兩者都是收斂狀態，其正確率也約99%，確實這兩層Dropout對於這種單純且淺層的網路作用不大。

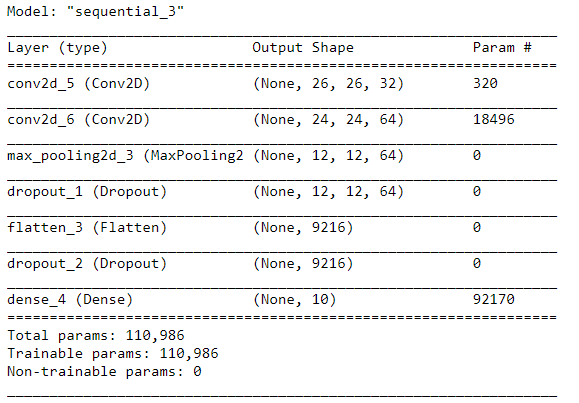
第二個問題與討論是紅箭頭處：



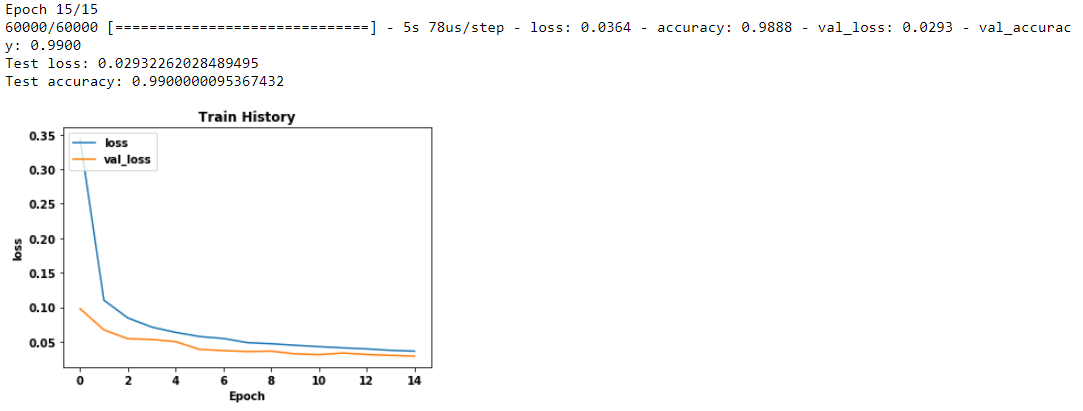
紅框以及紅箭頭部分是修改的地方，紅箭頭的部份是刪除一層Dense，紅框部分為修改參數，於上次作業已討論過，本次討論紅箭頭的部分。一般來說簡單的CNN模型只會用到一層全連接層，會用到兩層FC可能有以下考量：

1. 減少參數量，龐大的網路例如YOLO、VGGNet，第一層FC參數量可能上億，透過第二層FC減少運算量以及記憶體占用。
2. 影響辨識率最主要還是Conv2D層，額外再增加FC效果有限，因此大部分龐大的模型都是用2~3層FC，甚至有些會用1X1的Conv2D代替所有的FC層。

綜觀以上兩點，於此實作模型過於淺層簡單，多一層Dense效果應是差不了多少。



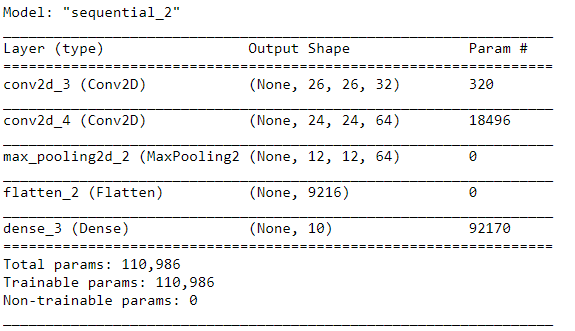
(原模型刪除第一層Dense)



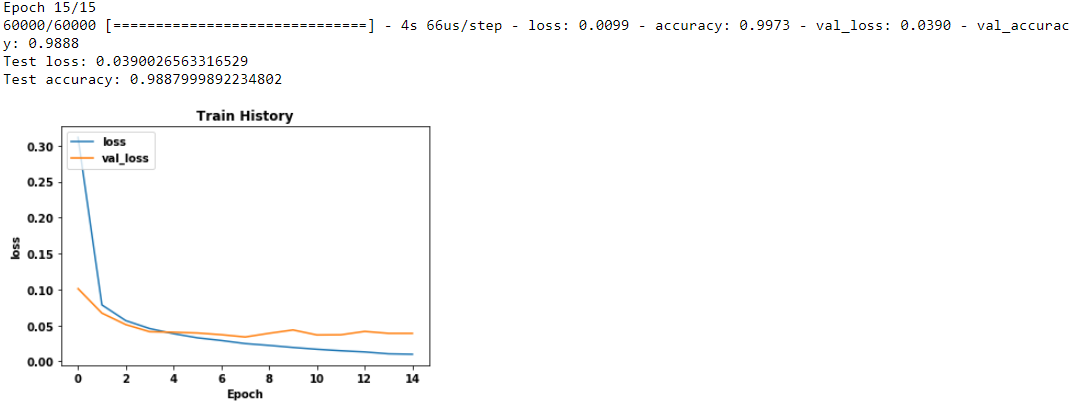
(原模型刪除第一層Dense)

可以見到結果確實如前面所述，效果與原本模型差不多，Loss=0.03、acc=99%。

再來是將兩種情形結合，模型簡化，去除第一層Dense與兩層Dropout。



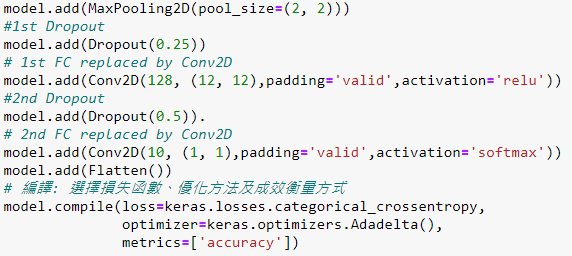
(去除第一層Dense與兩層Dropout)



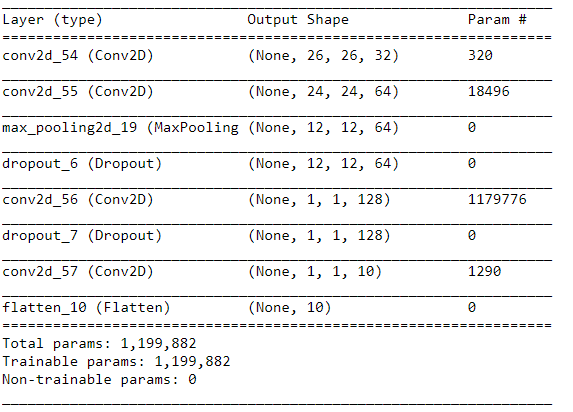
(去除第一層Dense與兩層Dropout)

可以發現訓練時間快了一些，原本76us->66us，速度提升13%，其代價是正確率下降0.2%左右。

前面紅字所述，有些模型會用1X1的Conv2D代替FC層，以下實作：

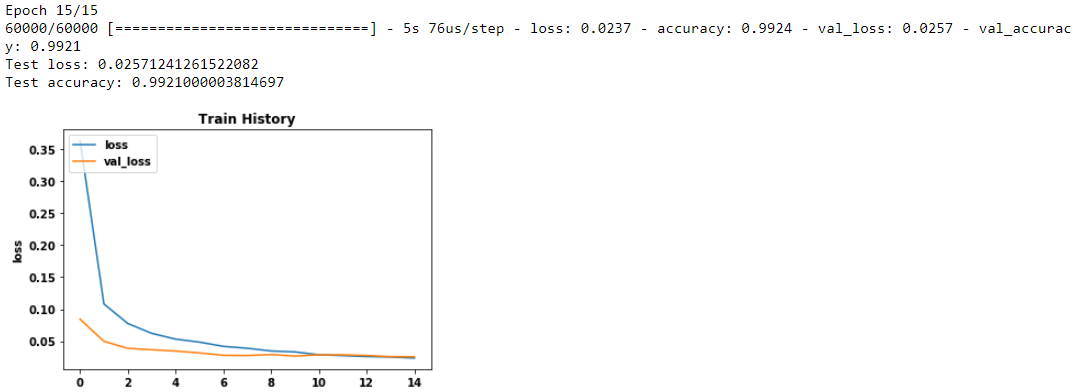


僅更動後兩層Dense替換成Conv2D，其餘皆未變動。

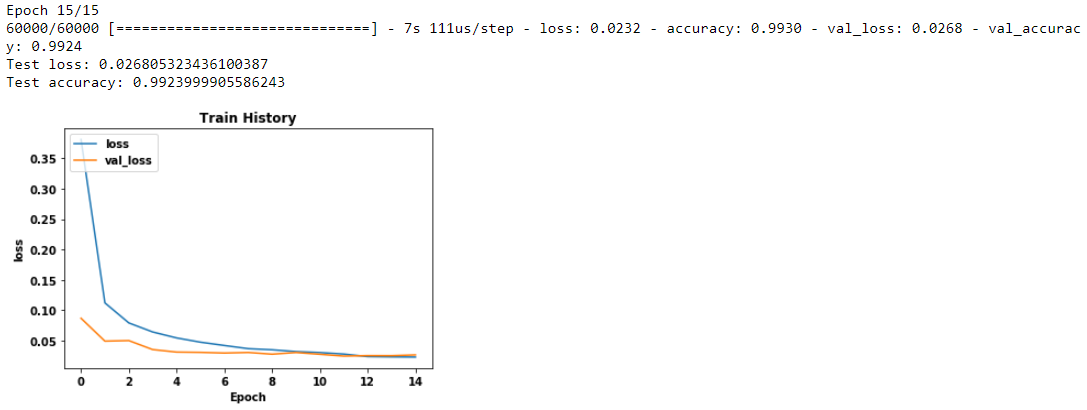


(two FC layers replaced by Conv2D)

\*\*其參數量與未更改前的模型配置相同



(尚未修改前的模型)



(two FC layers replaced by Conv2D)

觀察兩者其實沒有甚麼太大的差異，具體而言，將最後的全連接層替換層卷積層是不適當得根據不同應用決定，例如在此實例中，替換成卷積層效果並無提升太多，但耗時大大增加。

原本耗時76us->111us，增加近45%時間，非常耗資源，這也顯示出卷積層對計算負擔比全連接層還要重許多。近年來發展出另一個取代FC的做法：Global Average Pooling(GAP)，這些做法其根本目的都是要減少參數以降低overfitting。

實作中較大的問題是該如何更動架構、加加減減甚麼層，因為盲目的一直增加Conv2D沒有甚麼意義，而且到了一定程度模型會爛掉，因此嘗試了一些新奇的玩意。在撰寫報告時有一段是大部分的CNN模型使用的FC是幾層，於前次作業中個人斷定2層Dense是多餘的，僅需要一層，但深入查找資料後發現VGG

使用高達3層Dense，也是目前看過最多FC的模型，YOLO為2層，但這些是很複雜的模型，因此對於實作這種簡單的模型還是1層FC就夠了。於一篇[stackoverflow](https://stackoverflow.com/questions/47007658/idea-behind-how-many-fully-connected-layers-should-be-use-in-a-general-cnn-netwo)看到有人回復：某些模型會將FC使用卷積層代替。額外參考了[兩](https://medium.com/@mein2work/converting-fc-layers-to-conv-layers-8a43880a44ed)[篇](https://stackoverflow.com/questions/62235453/convert-fully-connected-layer-to-conv2d-and-predict-output)文章，因為自行修改架構總是出現維度對不上的問題，參考第二篇後才知道需額外添加一層Flatten。

在本次作業做了許多架構更動的部分，基本上不會再增加層數，不僅會浪費時間，效果提升也不會很明顯，但替換成其他layer也能發現，效果有限，畢竟模型實在是不夠複雜。尋找參考資料時第一次知道可以將全連接層替換成卷積層甚至GAP，深入實作發現這個方法好像頗雞肋，也許效果有稍微提升，但是增加的時間實在是不成比例，難怪會再發展出GAP，具體來說也還不知道為何要使用Conv替代FC，個人認為是為了減少權重參數量，避免overfitting。

[Code link](https://github.com/autotntfan/Design-practice-and-applications-of-machine-learning/blob/master/L3/DPPML_HW3_B063012054_gpu.ipynb)