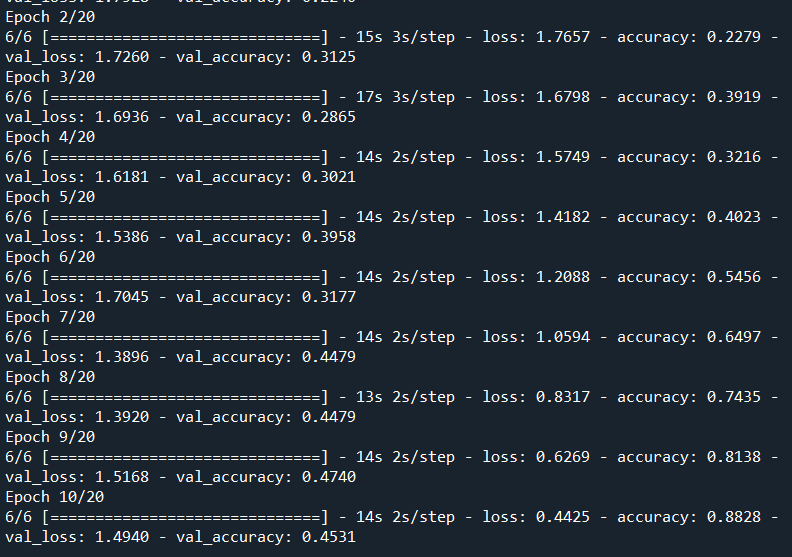
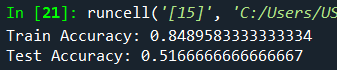
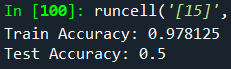
**HW5 REPORT 109070032廖品睿**

1.First time training model with original data we stop at the 10th iteration and only get a total accuracy of 0.8828, which has a train accuracy of 0.8489 and a test accuracy of 0.5166

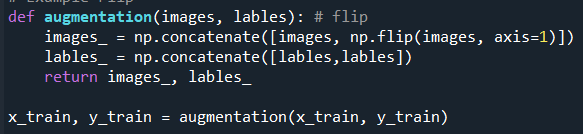




2.The second time we trained the model with flipped data (which has 1200\*0.8\*2=1920 pictures), we stop at the 8th iteration and we get a total accuracy of 0.9052, which has a train accuracy of 0.9781 and a test accuracy of 0.5



(a) I used np.flip to double the input data into 1920 pictures to help prediction



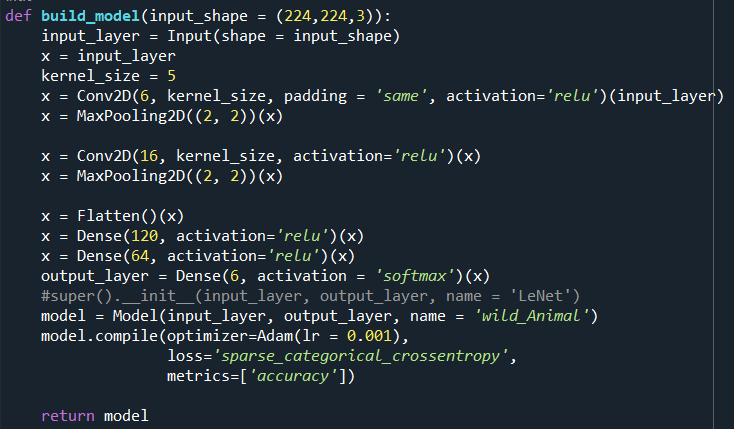
(b)The model consists of about 4 parts

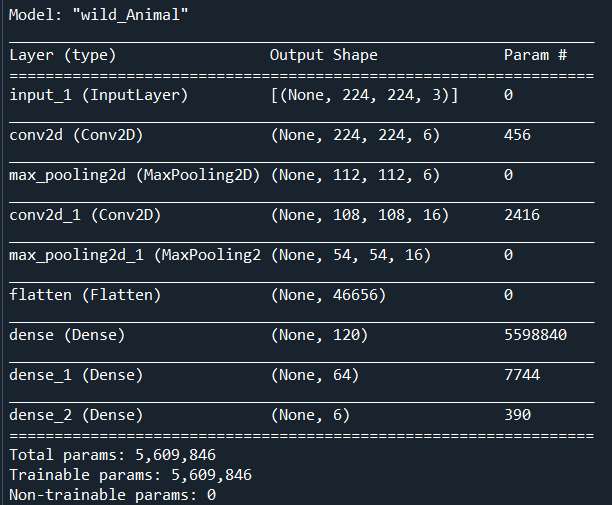
The first part is overlap the pictures together and get values to send it into the convolutional layer, make the kernel size a 5x5 box that scans through the map , also making the padding margin default value all 0.

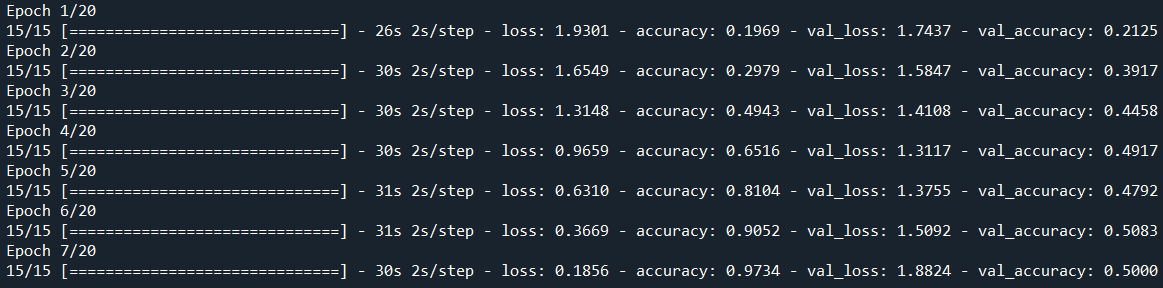
The second part is to do pooling and fill in the max values in a 2x2 square, and repeat the process two times.

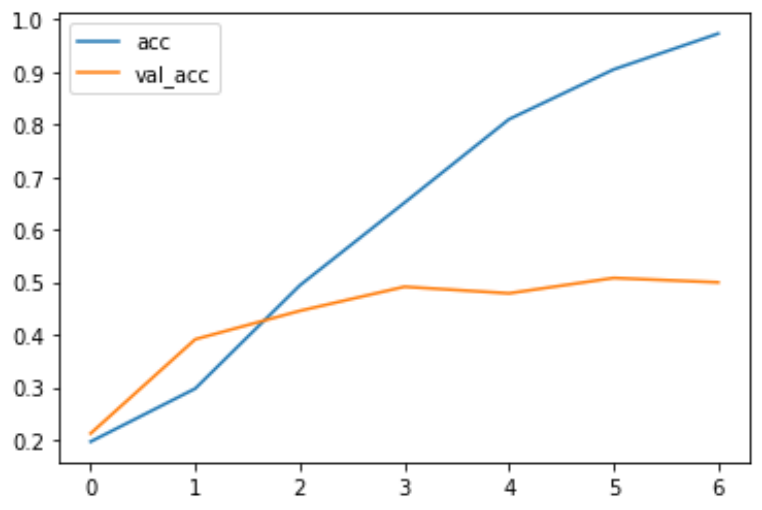
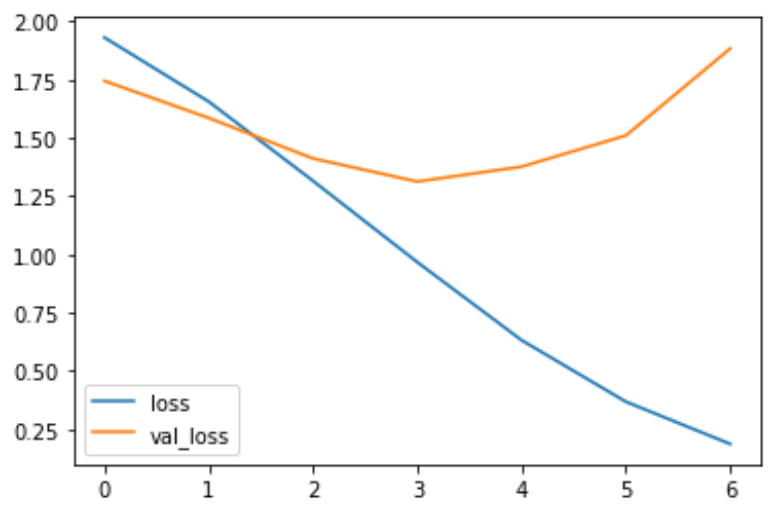
The third part is to flatten the characteristics into a series of nodes, and create more layer with decreasing numbers of nodes 120-->64-->6, which processes its probability through relu and softmax function.

Finally, we use the Adam optimizer with 0.001 learning rate to compile and learn over the loss.









1. 如何優化模型的方法

I think we can add convolution or pooling layer or increase the number of hidden layers in the neural network to optimize the model. Also, we can adjust the kernel size, batch size, max pooling size and learning rate to try to optimize model.

1. 結果分析

Tried adjusting many parameters like making batch size=16 or kernel size=7, turns out that it only causes overfit and does hot help increasing model accuracy, after trying the data generator , I found that increase the numbers of data is the most effective way of increasing model accuracy.

The final best result is at the bottom.

3. 遇到的問題或其他心得

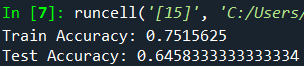
I initially could not append the new data in a short time to the original x\_train data, at last I used the concatenate function to connect new data to the old ones.

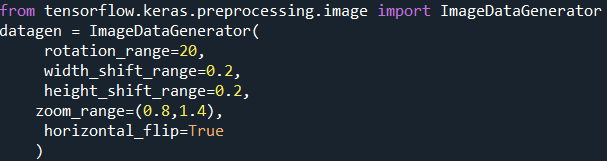
Found the question that using the function early stopping could cause underfitting so I set a fixed epoch=40 to train the model better.

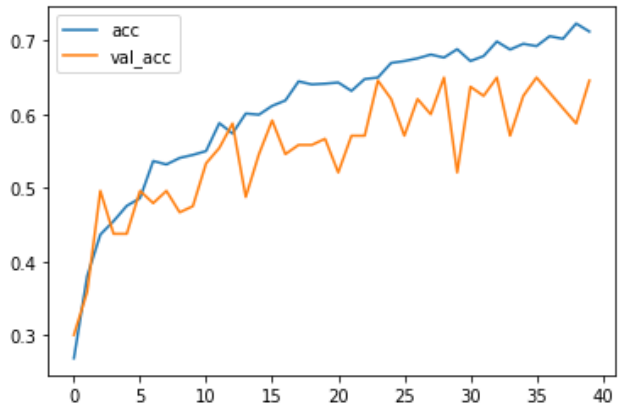
**Best model here:**

Data generator + epoch=40

Here It makes the train acc about 0.65 (better)

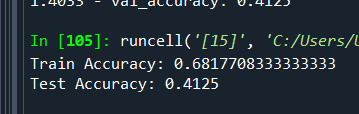




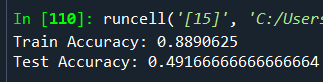


Other Experiments :

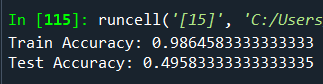
Lr=0.0001



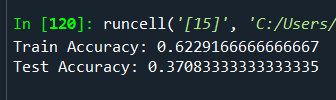
Kernel size=3



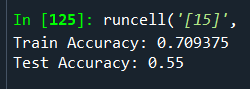
Kernel size=7



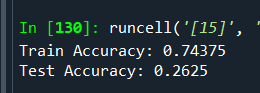
Kernel size=9



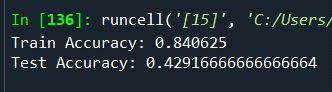
Max pooling 4x4



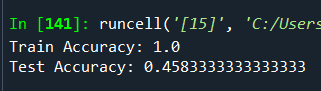
Batch normalization:



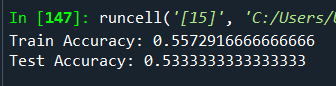
addx = Dense(64, activation='relu')(x)



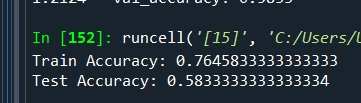
Batch size=16



Add data generator



Only zoom in data generator



Zoom + horizontal flip

