

HW4_E94076089_謝獻霆

1. Model :

i. Architecture

只使用兩層 convolution，其中用 batch_normalization 與 dropout 防止 overfitting，並用 Prelu 使負值部分的斜率是根據數據來訂。

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 310, 310, 32)	4736
batch_normalization_1 (Batch Normalization)	(None, 310, 310, 32)	128
p_re_lu_1 (PReLU)	(None, 310, 310, 32)	3075200
dropout_1 (Dropout)	(None, 310, 310, 32)	0
max_pooling2d_1 (MaxPooling2D)	(None, 155, 155, 32)	0
conv2d_2 (Conv2D)	(None, 155, 155, 64)	51264
batch_normalization_2 (Batch Normalization)	(None, 155, 155, 64)	256
p_re_lu_2 (PReLU)	(None, 155, 155, 64)	1537600
dropout_2 (Dropout)	(None, 155, 155, 64)	0
max_pooling2d_2 (MaxPooling2D)	(None, 77, 77, 64)	0
flatten_1 (Flatten)	(None, 379456)	0
dropout_3 (Dropout)	(None, 379456)	0
dense_1 (Dense)	(None, 1024)	388563968
p_re_lu_3 (PReLU)	(None, 1024)	1024
batch_normalization_3 (Batch Normalization)	(None, 1024)	4096
dropout_4 (Dropout)	(None, 1024)	0
dense_2 (Dense)	(None, 3)	3075
Total params: 393,241,347		

2. Preprocessing :

i. Method:

將資料分成三批，並依序對這些圖片做訓練。

第一種：原始圖片 5200 張

第二種：去除背景後的照片 5200 張

第三種：去除背景後再轉成黑白的照片 5200 張

剩下 400 張為 validation

並統一經過下面的 ImageDataGenerator，轉成(310,310,3)的形式，

我想說圖片越大越容易找到特徵

```
datagen = ImageDataGenerator(  
    rotation_range=40,  
    horizontal_flip=True,  
    vertical_flip=True,  
    fill_mode='nearest',  
    rescale=1./255.  
)
```

ii. Example:

00018. jpg

第一張圖的預處理結果



00018. jpg

第二張圖的預處理結果



00018. jpg

第三張圖的預處理結果



00010. jpg

第四張圖的預處理結果



3. Result :

i. Evaluation with training dataset :

```
Epoch 1/15
485/485 [=====] - ETA: 0s - loss: 1.2859 - accuracy: 0.4676
Epoch 00001: val_loss improved from inf to 1.03592, saving model to Pepita.h5
485/485 [=====] - 6047s 15s/step - loss: 1.2859 - accuracy: 0.4676 - val_loss: 1.0359 - val_accuracy: 0.4543 - lr: 0.0050
Epoch 2/15
485/485 [=====] - ETA: 0s - loss: 1.2341 - accuracy: 0.4639
Epoch 00002: val_loss did not improve from 1.03592

Epoch 00002: ReduceLROnPlateau reducing learning rate to 0.00249999999441206455.
485/485 [=====] - 3426s 8s/step - loss: 1.2341 - accuracy: 0.4639 - val_loss: 1.1347 - val_accuracy: 0.4038 - lr: 0.0050
Epoch 3/15
485/485 [=====] - ETA: 0s - loss: 1.0681 - accuracy: 0.5009
Epoch 00003: val_loss improved from 1.03592 to 0.92324, saving model to Pepita.h5
485/485 [=====] - 418s 1s/step - loss: 1.0681 - accuracy: 0.5009 - val_loss: 0.9232 - val_accuracy: 0.5769 - lr: 0.0025
Epoch 4/15
485/485 [=====] - ETA: 0s - loss: 1.0163 - accuracy: 0.5321
Epoch 00004: val_loss did not improve from 0.92324

Epoch 00004: ReduceLROnPlateau reducing learning rate to 0.0012499999720603228.
485/485 [=====] - 3408s 8s/step - loss: 1.0163 - accuracy: 0.5321 - val_loss: 0.9436 - val_accuracy: 0.5192 - lr: 0.0025
Epoch 5/15
485/485 [=====] - ETA: 0s - loss: 0.9551 - accuracy: 0.5486
Epoch 00005: val_loss improved from 0.92324 to 0.88337, saving model to Pepita.h5
485/485 [=====] - 445s 1s/step - loss: 0.9551 - accuracy: 0.5486 - val_loss: 0.8834 - val_accuracy: 0.5697 - lr: 0.0012
Epoch 6/15
485/485 [=====] - ETA: 0s - loss: 0.9382 - accuracy: 0.5614
Epoch 00006: val_loss did not improve from 0.88337

Epoch 00006: ReduceLROnPlateau reducing learning rate to 0.0006249999860301614.
485/485 [=====] - 3373s 8s/step - loss: 0.9382 - accuracy: 0.5614 - val_loss: 0.9199 - val_accuracy: 0.5625 - lr: 0.0012
Epoch 7/15
485/485 [=====] - ETA: 0s - loss: 0.9026 - accuracy: 0.5809
Epoch 00007: val_loss did not improve from 0.88337

Epoch 00007: ReduceLROnPlateau reducing learning rate to 0.0003124999930150007.
485/485 [=====] - 82s 201ms/step - loss: 0.9026 - accuracy: 0.5809 - val_loss: 0.8991 - val_accuracy: 0.5625 - lr: 6.2500e-04
Epoch 8/15
485/485 [=====] - ETA: 0s - loss: 0.8888 - accuracy: 0.5769
Epoch 00008: val_loss improved from 0.88337 to 0.88218, saving model to Pepita.h5
485/485 [=====] - 1s 988ms/step - loss: 0.8888 - accuracy: 0.5769 - val_loss: 0.8822 - val_accuracy: 0.5697 - lr: 3.1250e-04
```

Train Loss: 0.9026

Train Accuracy: 0.5809

ii. Evaluation with testing dataset :

Test Loss: 0.5991

Test Accuracy: 0.6025

iii. Confusion matrix :

		Predict		
		A	B	C
LABEL	A	21	96	22
	B	7	343	38
	C	4	151	118