



CXR8 – CNN WITH KERAS



蔡昕翰 葉安之 周楷蓉

指導老師：李柏翰

簡介

我們在醫療科技展上看到了長庚大學做了以腕部X光辨識骨質疏鬆的AI，讓我們著迷不已。
這讓我們想到，說不定能用卷積神經網路(CNN)辨識胸部X光。

熱度圖顯示判讀依據

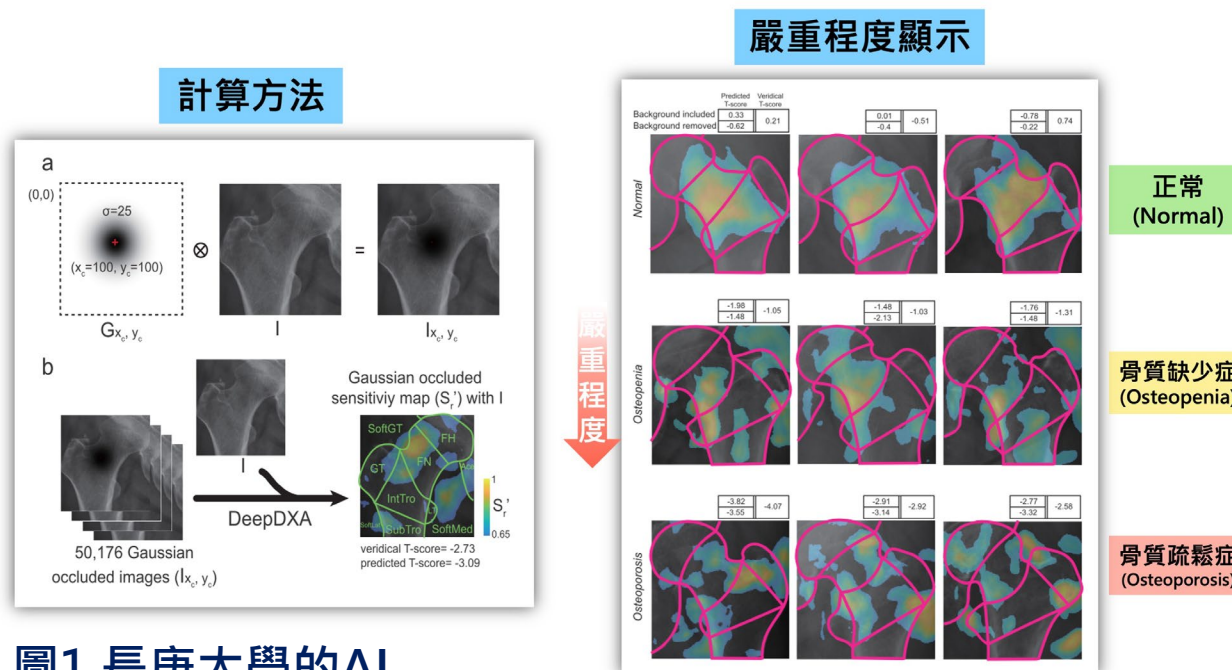


圖1 長庚大學的AI

參展照片

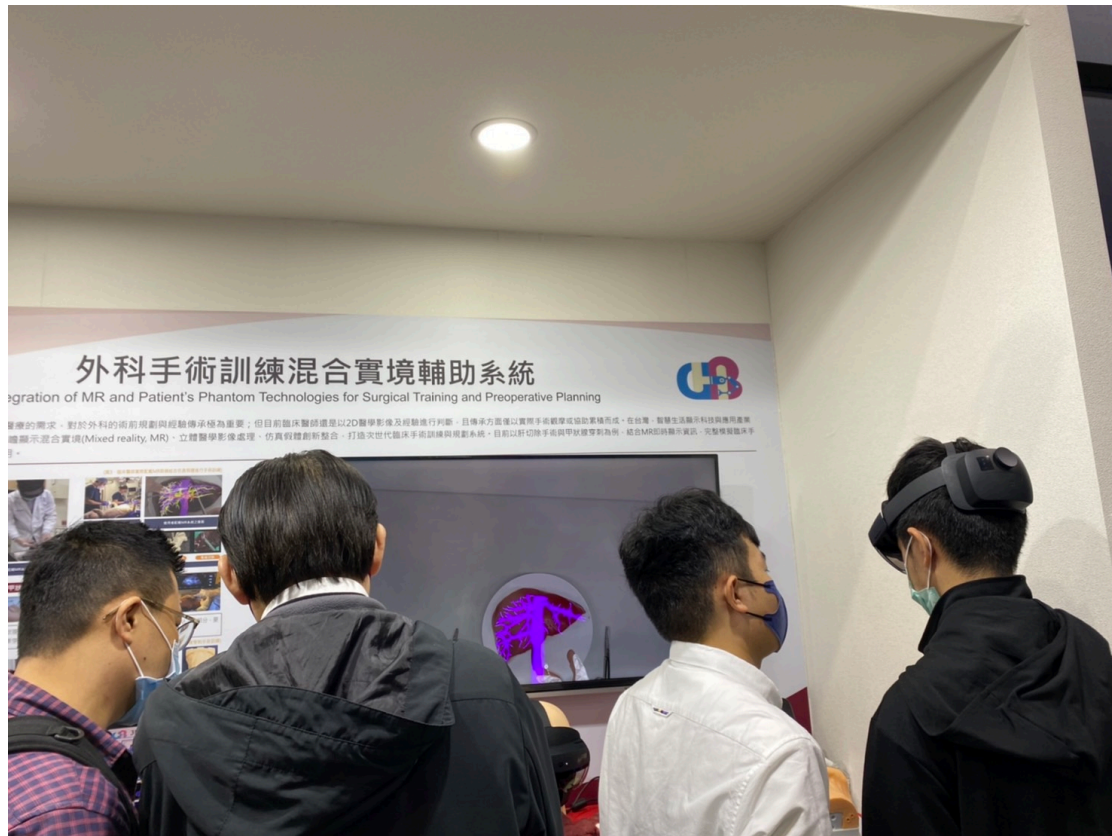


圖 2-1 我們參展看到的事物



圖 2-2 三校聯盟

CXR8介紹

- 此資料集收集了約11萬張胸腔X光照片
- 含19種異常現象和1個無異常的標籤
- 每張照片都是1024x1024

表 1 醫學名詞翻譯

Atelectasis	Cardiomegaly	Consolidation	Edema	Effusion	Emphysema	Fibrosis	Hernia	Infiltration	Mass
肺塌陷	心臟肥大	實變	水腫	肋膜積水	肺氣腫	纖維化	疝氣	肺浸潤	腫塊
Nodule	Pleural Thickening	Pneumonia	Pneumothorax	Pneumoperitoneum	Pneumomediastinum	Subcutaneous Emphysema	Tortuous Aorta	Calcification of the Aorta	No Finding
結節	胸膜增厚	肺炎	氣胸	腹腔積氣	縱膈腔氣腫	皮下氣腫	主動脈弓彎曲	動脈粥狀硬化	無異常



圖 3 CXR8內的資料

模型基本架構

- 使用CNN(卷積神經網路)
- 優化器使用SGD
- 損失函數用MSE與MAE

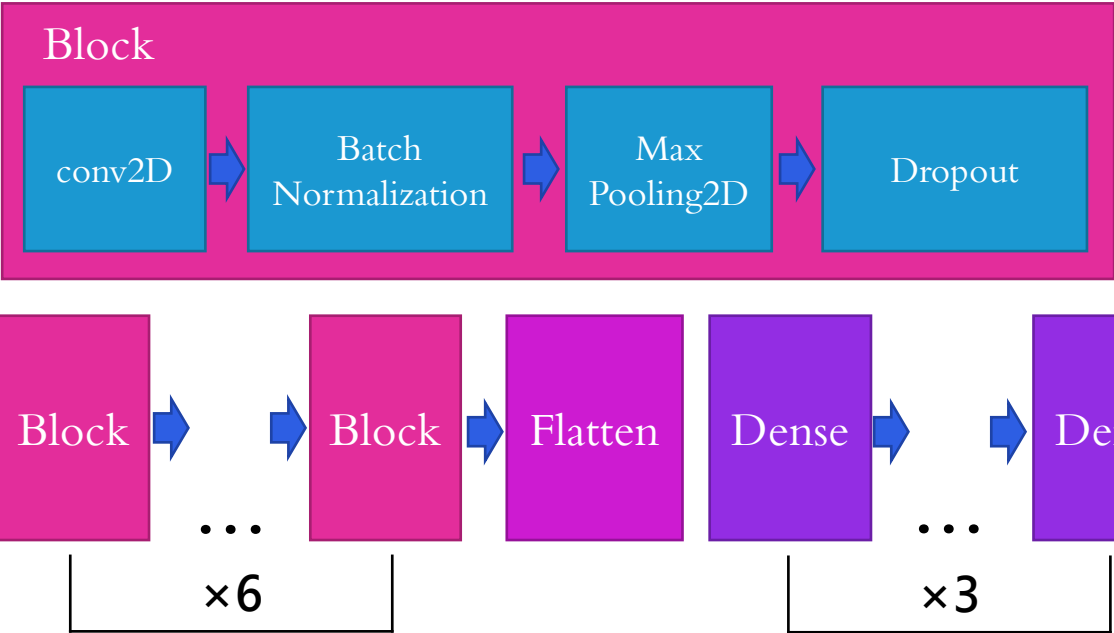


圖 4-1 模型示意圖

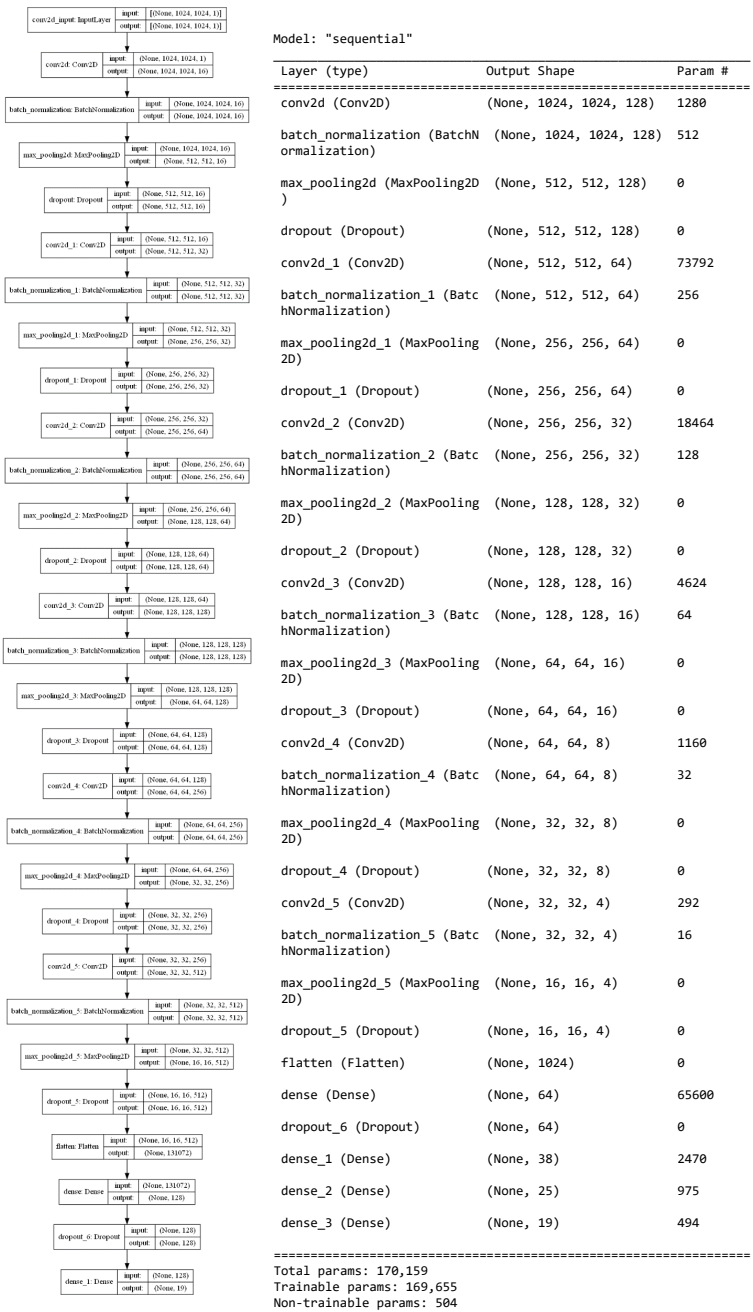


圖 4-2 完整模型圖

訓練資料

- 我們使用此資料集的前640個資料訓練，下160個用來評估，太多會GPU記憶體不足
- Batch Size設為2，太大會GPU記憶體不足
- 訓練50次
- 標籤取前19個，即把無異常拿掉

```
EPOCHS = 50
BATCHSIZE = 2
checkpoint_filepath = 'ckpts/checkpoint_batchnormalized-mae.h5'
model_checkpoint_callback = tf.keras.callbacks.ModelCheckpoint(
    filepath=checkpoint_filepath,
    save_weights_only=True,
    monitor='val_accuracy',
    mode='max',
    save_best_only=True)
history = model.fit(x=x_train[:800], y=y_train[:800], batch_size=BATCHSIZE, epochs=EPOCHS,
    callbacks=[model_checkpoint_callback], validation_split=0.2, verbose=1)
```

圖 5 訓練的程式

損失函數MSE

- 可以看到有一點overfit。
- 準確度僅40%

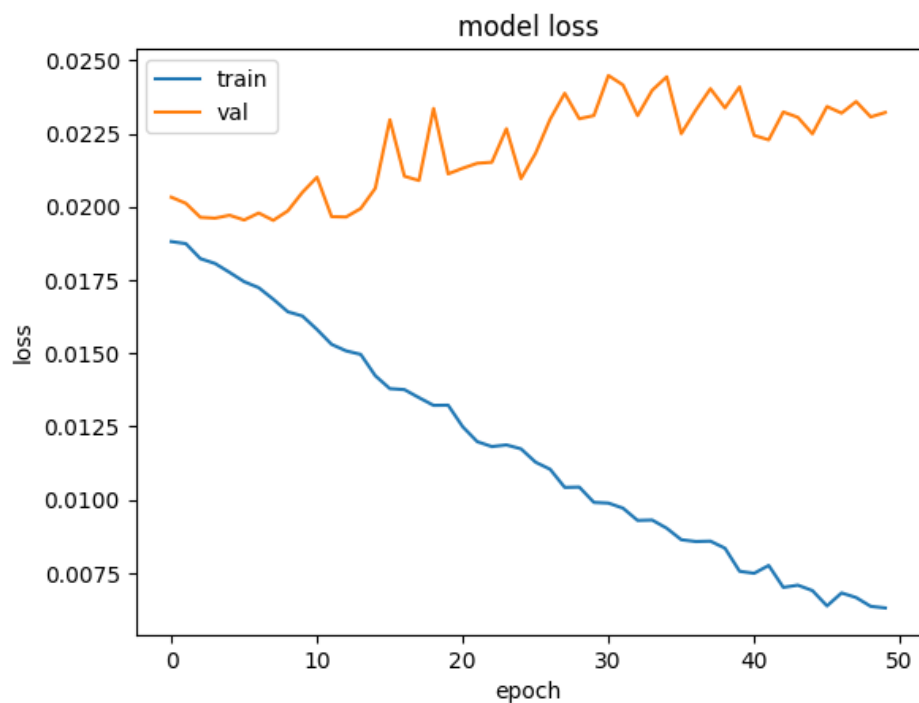


圖 6-1訓練Loss記錄

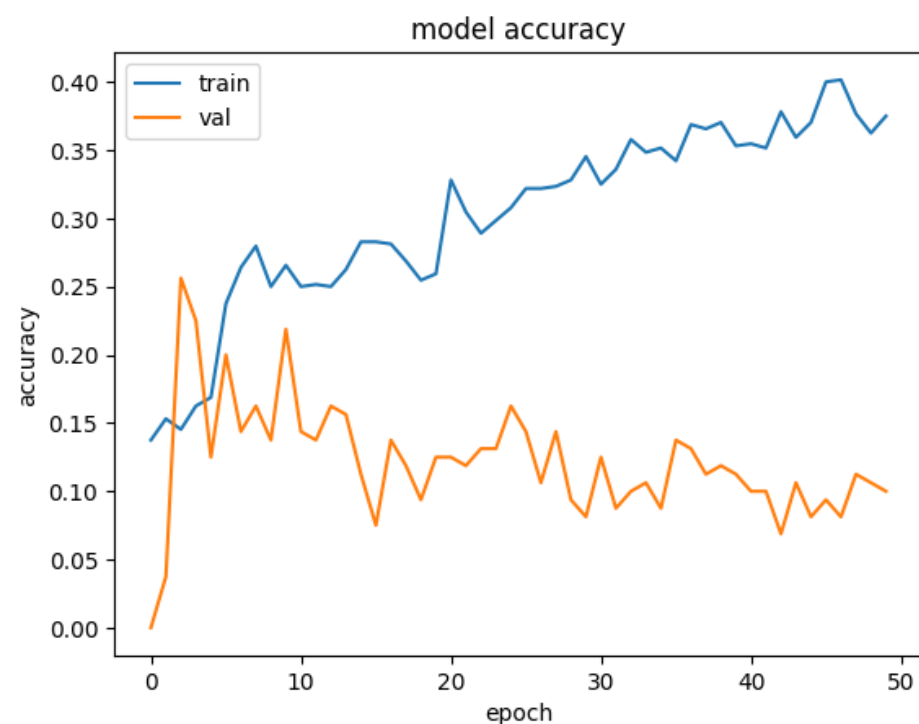


圖6-2 訓練accuracy記錄圖

MSE confusion matrix

- 可以看到，就是一個很不準的網路。
- 需要更多EPOCH
- 明明有19個label，卻只出現17個

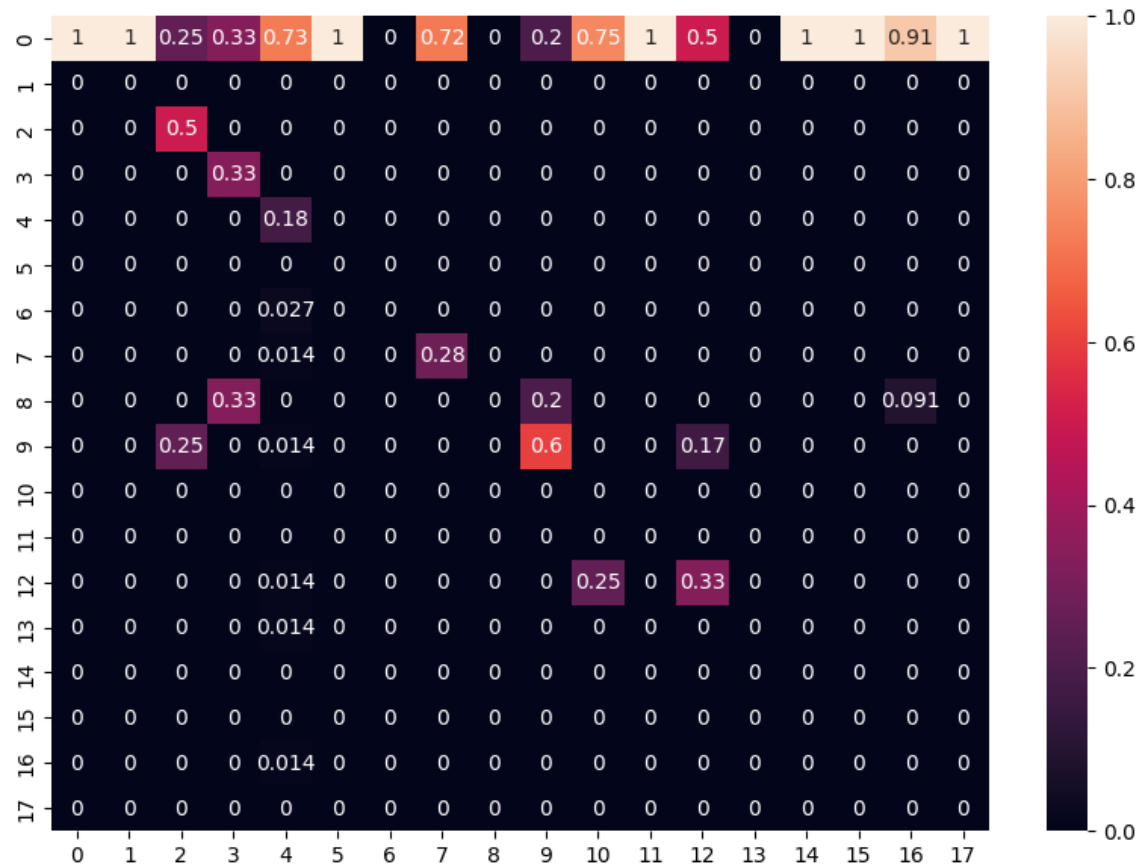


圖 6-3 混淆矩陣

可能問題(不一定是對的)

- LOSS可能太小了，因為我們資料差一定小於1，平方後就更小了。可以用MAE試試看。
- 資料集裡每項分布不均，其中還有兩項只有0個，導致confusion matrix只有17項
- Softmax函數每項加起來是1，但把無異常標籤拿掉後，會導致有些資料全0

```
In [48]: print(np.sum(y_train[:800, ...], axis=0))
```

```
[44.  7. 13.  3. 36.  0.  5.  3. 84. 23. 21.  7.  0. 17.  1.  1.  1.  8.
 1.]
```

圖 7 分布不均的資料

MAE

- 模型沒有在學習
- 他預測的東西全部都一樣(以訓練資料集前200個測試)
- 可能因為ReLU左邊都是0，導致無法激勵某些Node

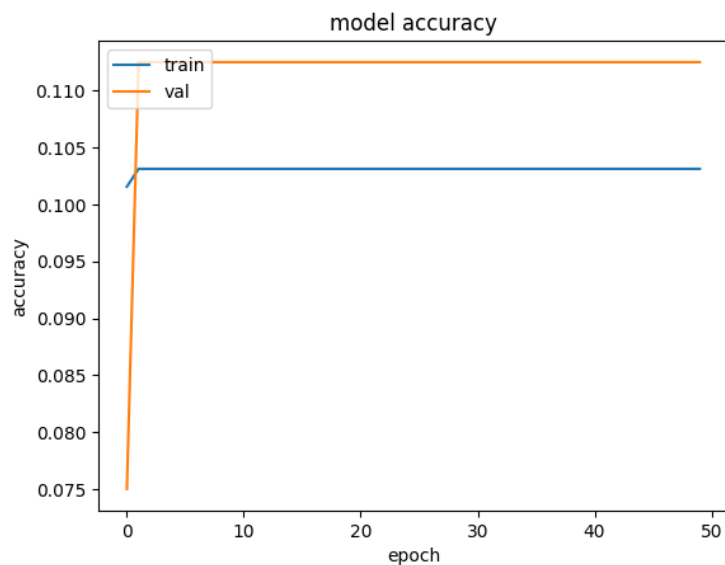


圖 8-1 accuracy

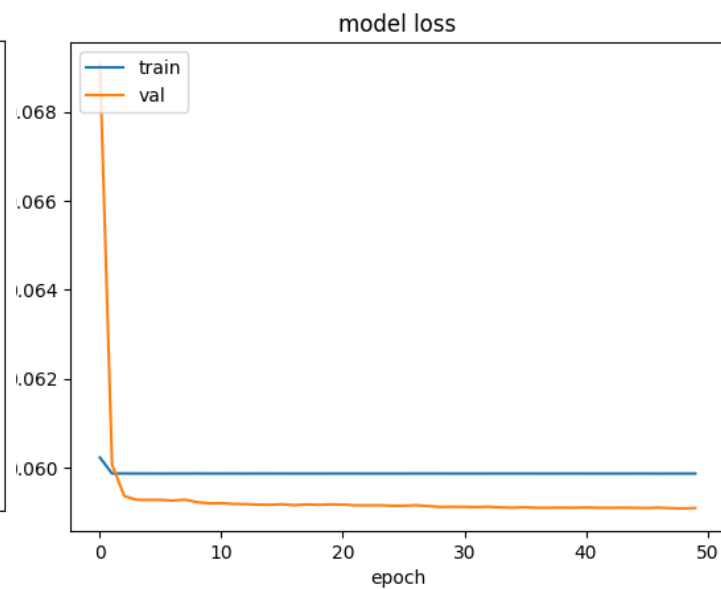


圖8-2 Loss

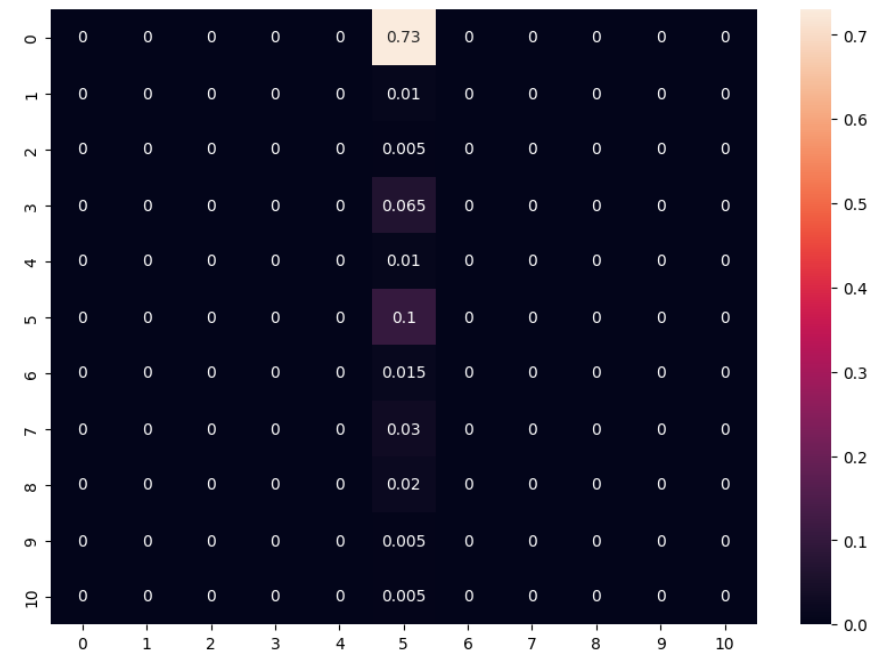


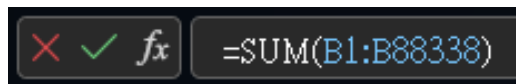
圖 8-3 confusion matrix

重新選擇資料

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
4003	1034	1234	599	3804	315	689	89	9192	2068	2602	1068	298	2004	103	40	70	446	185	58494

圖 9 Excel分析結果

1. 用excel看資料分布



2. 看到40是最小的，所以每項都取40個

3. 用Python將資料分出來

```
import pandas as pd
from sklearn.utils import shuffle

# constants
ALL_IN_LABEL = "label.csv"
TRAIN_OUT_PATH = "train_label.csv"
TEST_OUT_PATH = "test_label.csv"
test_split = 0.25

df_in = pd.read_csv(ALL_IN_LABEL) # read all labels
df_in = shuffle(df_in) # shuffle

# declare Outputs
df_out_train = pd.DataFrame(columns=df_in.columns)
df_out_test = pd.DataFrame(columns=df_in.columns)

# find the minimum sum of the rows
minimum = df_in.drop(["subject_id", "id"], axis=1).sum(numeric_only=True, axis="rows").min()
split_point = int(test_split * minimum) # where should it cut

for c in df_in.columns[:-1]:
    # filters data that its column c is 1 and concatenates it to output
    df_out_test = pd.concat([df_out_test, df_in[df_in[c] == 1].iloc[:split_point, ...]])
    df_out_train = pd.concat([df_out_train, df_in[df_in[c] == 1].iloc[split_point:minimum, ...]])

# shuffles the datas
df_out_train = shuffle(df_out_train)
df_out_test = shuffle(df_out_test)

# save to disk
df_out_train.to_csv(TRAIN_OUT_PATH, index=False)
df_out_test.to_csv(TEST_OUT_PATH, index=False)
```

圖 10 分資料程式

重新定義model

- 使用Categorical Cross Entropy
- 使用SGD
- Dropout從0.1升至0.4
- Batch Size從2升至10
- 這次把CNN blocks替換為一個MobileNet

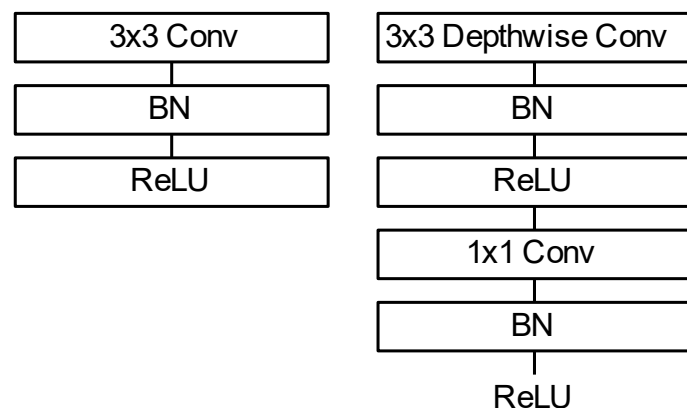


圖 11-1一般Conv2D與DepthwiseConv2D差異

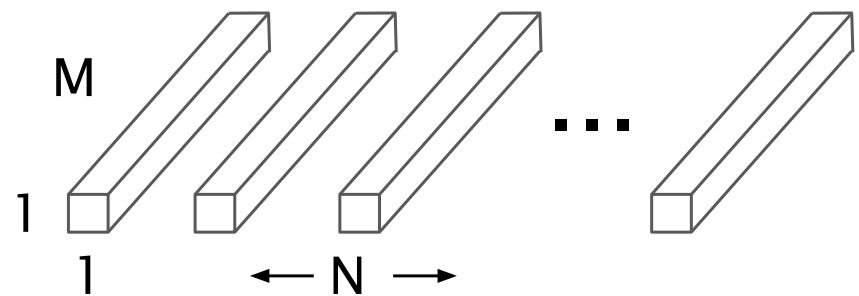
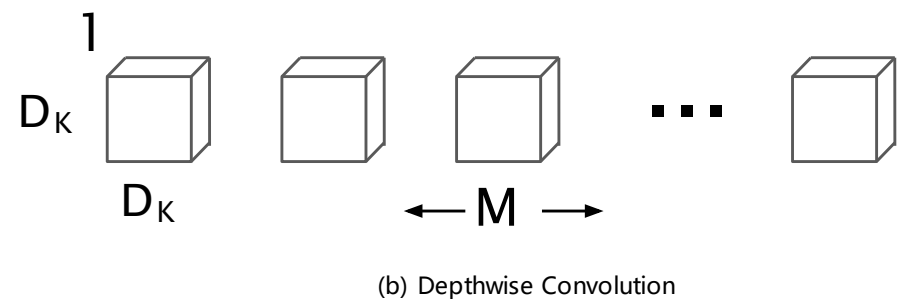
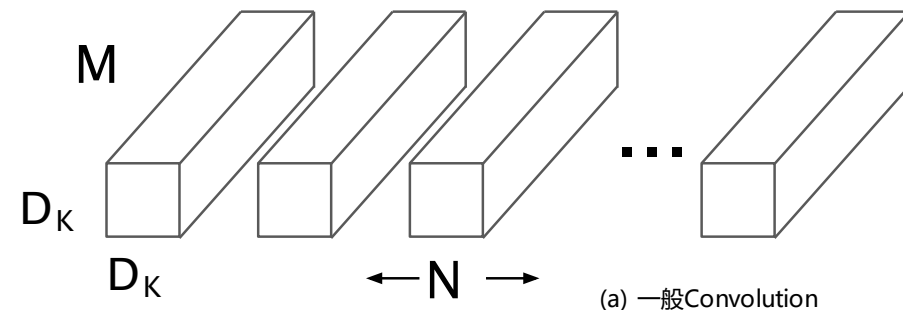


圖11-2 一般CNN與MobileNet差異

MobileNet

- Loss下降到3左右就不下降了
- accuracy沒有變好，代表overfitting
- 他只會預測其中幾個

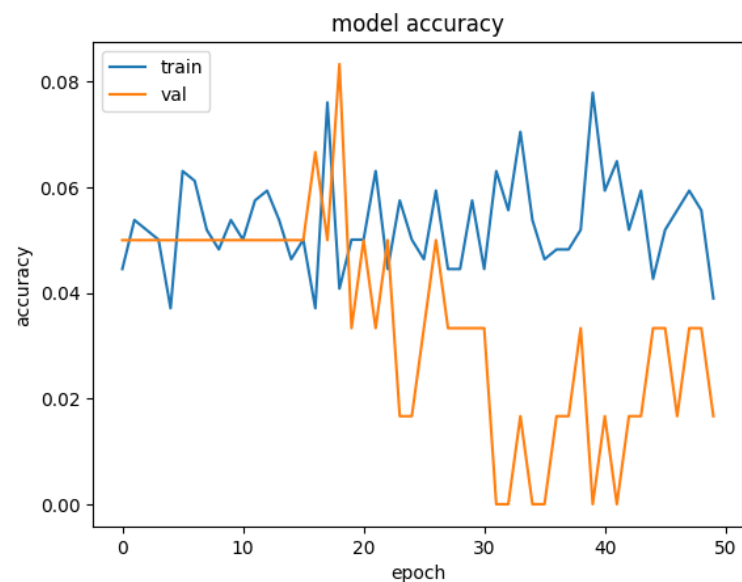


圖 12-1 accuracy

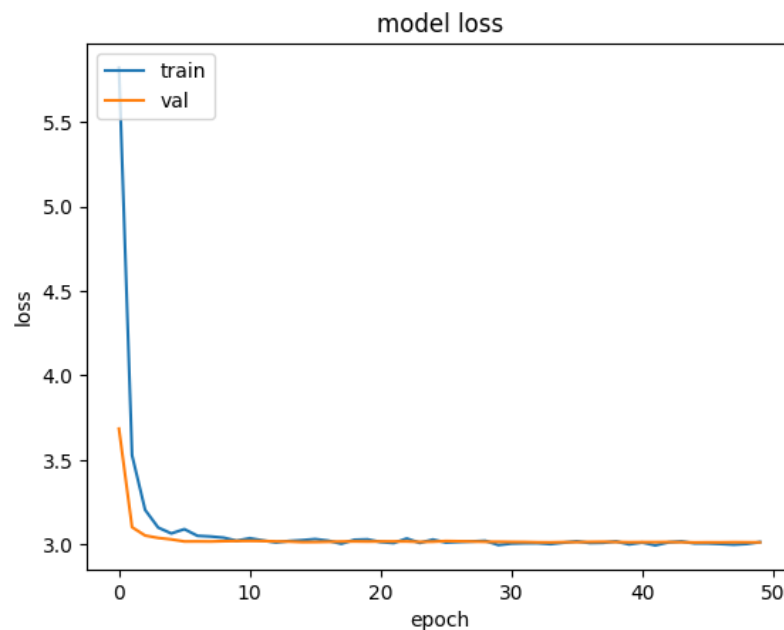


圖 12-2 Loss

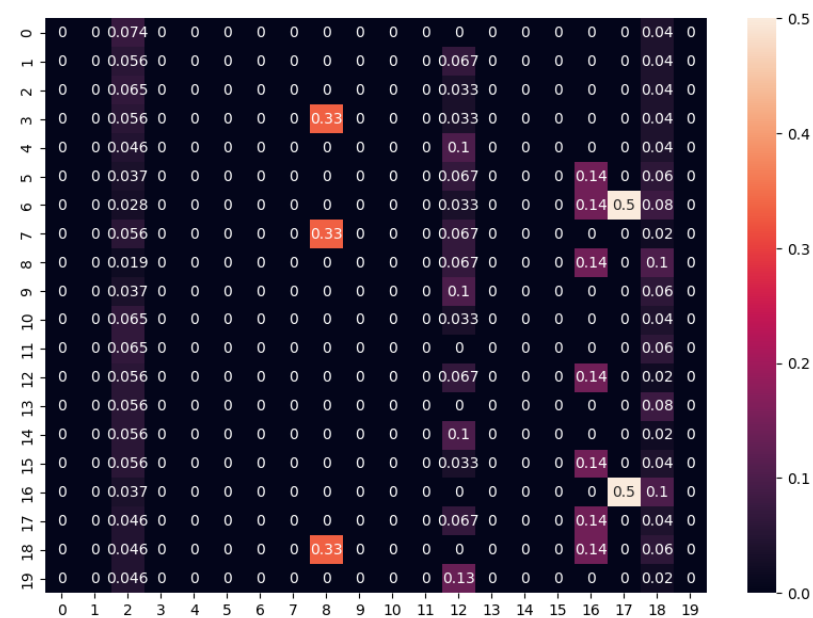


圖 12-3 confusion matrix

如何改進

- 想辦法讓資料變多變平均
- 或捨棄幾項太少的，例如只有40個的。
- 可以多試一些其他normalization方法
- 測試看看ResNet、EfficientNet等其他架構是否有辦法能提升訓練成績。

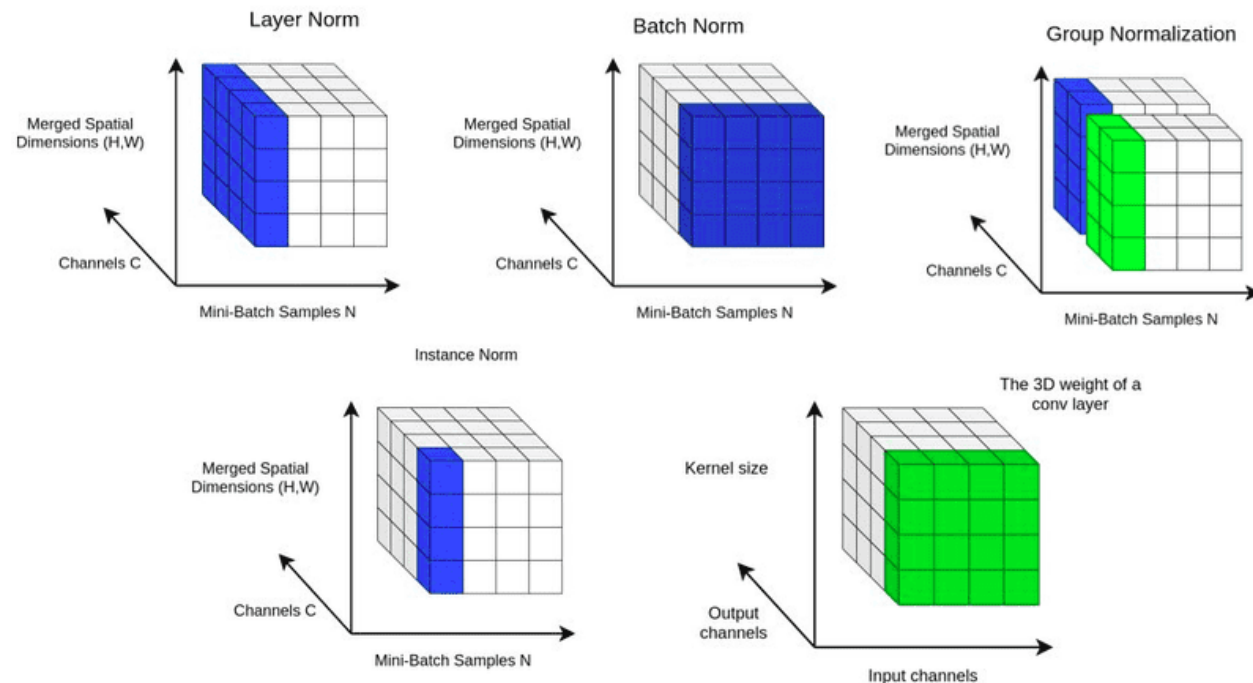


圖 13 不同的Normalization

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
4003	1034	1234	599	3804	315	689	89	9192	2068	2602	1068	298	2004	103	40	70	446	185	58494

圖 14 捨棄資料量太少的

參考資料

1. 人工智慧篩檢X光影像之骨質疏鬆風險 https://innoaward.taiwan-healthcare.org/award_detail.php?REFDOCTYPID=0oixvpdnj4w075ah&NumID=0qlrq0df3zsbqdod&REFDOCID=0qls2vjc3nnwf4vz&Num=1
2. In-layer normalization techniques for training very deep neural networks <https://theaisummer.com/normalization/>
3. CXR8 <https://nihcc.app.box.com/v/ChestXray-NIHCC>
4. Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv e-prints*, arXiv:1704.04861.