

# 1. Data

## 1.1. Dataset

此作業使用的資料集為胸腔 X 光影像，取自 [1]。影像檔案格式為 JPEG，每張圖片皆標記屬於正常、病毒性肺炎或細菌性肺炎，在此將病毒性肺炎與細菌性肺炎歸為同屬肺炎一類。因此，總共有 4273 張肺炎與 1583 張為正常胸腔 X 光影像。

## 1.2. Exploratory data analysis

首先觀察影像大小分布，可以看到肺炎與正常組別影像的長寬比皆呈現接近線性縮放關係（如圖 1），若劃上趨勢線肺炎影像分布的斜率為 0.86，而正常胸腔影像為 0.99，後者影像長寬比較接近正方，經由縮放後應不會有過度變形的問題。

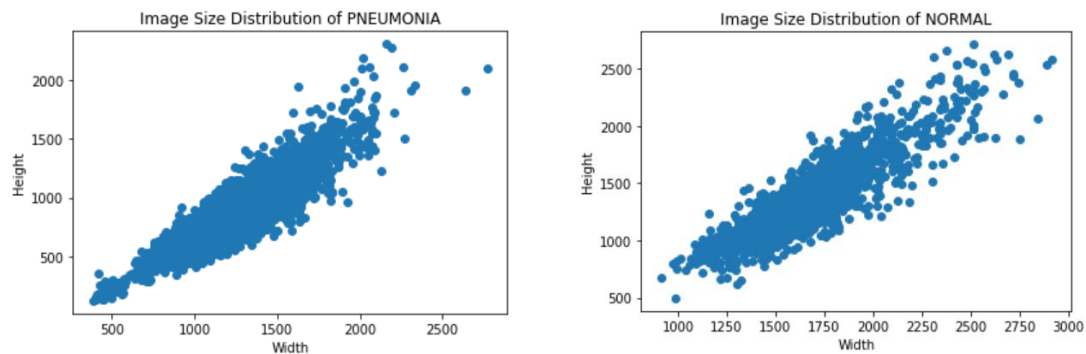


圖 1、影像大小分布圖

隨機從肺炎與正常組別取出 8 張原始影像觀察，可以發現正常組別的肺部輪廓較為清楚明顯，而肺炎組別的肺部輪廓較模糊，肺內部白色的部分較多且具有瀰漫性的白影，如圖 2。

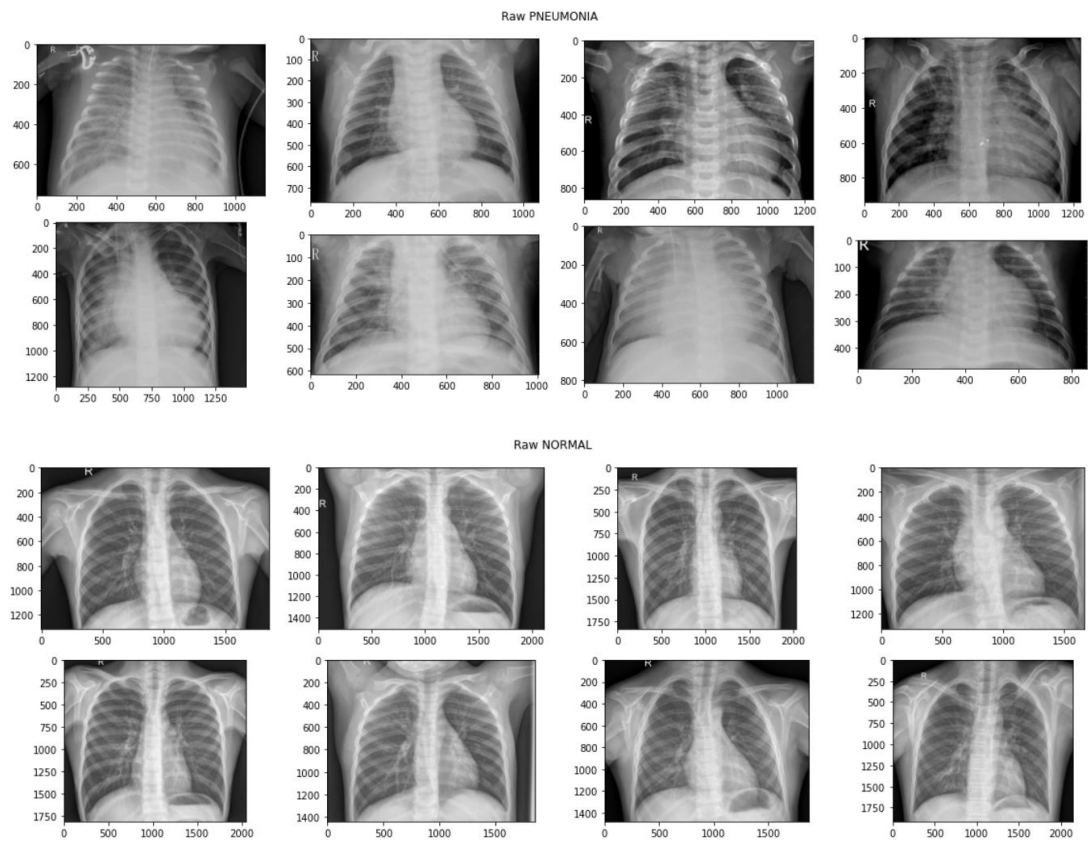


圖 2、肺炎與正常胸腔原始影像

將原始影像重新縮放成同一比例  $299 \times 299$ ，從肺炎與正常組別取出 8 張影像，如圖 3。與原始影像（圖 2）比較確定影像經由重新縮放後沒有過度變形而致無法判斷為肺部的問題，且仍保留原始影像的特徵，因此後續會將原始影像統一都縮放成  $299 \times 299$  後再輸入深度學習模型分析。

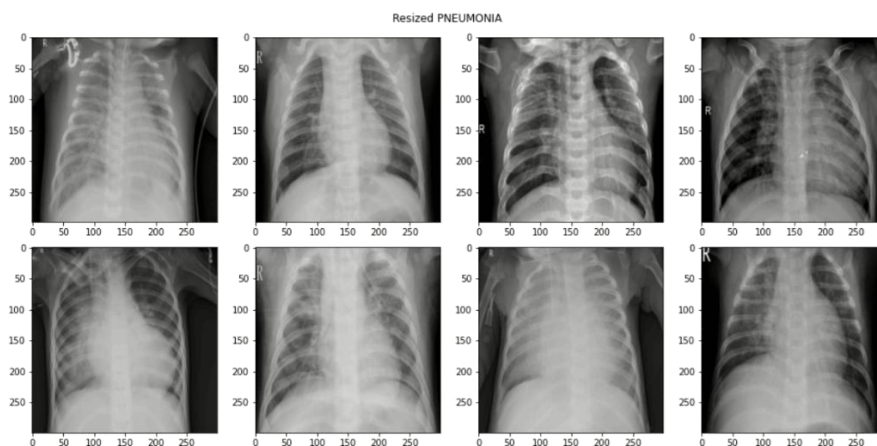


圖 3-a、縮放後的肺炎胸腔影像

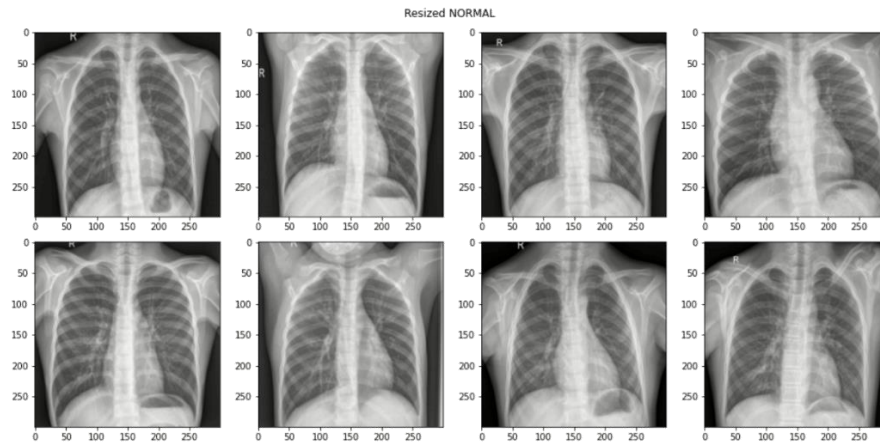


圖 3-b、縮放後的正常胸腔影像

將縮放後的正常與肺炎胸腔影像取平均後印出（如圖 4），可以發現正常組的肺部輪廓較為清晰，而肺炎肺部輪廓較為模糊且肺內部較白，與前面的觀察相同。將兩圖相減後（圖 5）可更明顯看出兩者的差異，肺部與外圍的部分因相差大於 0，表示肺炎影像的平均像素值較白，而肺以外的軀幹部分正常胸腔影像的平均像素值較白。

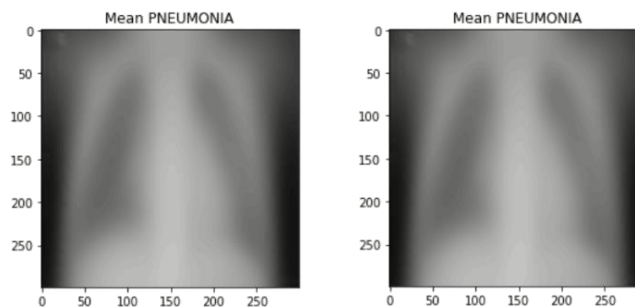


圖 4、肺炎與正常胸腔平均影像

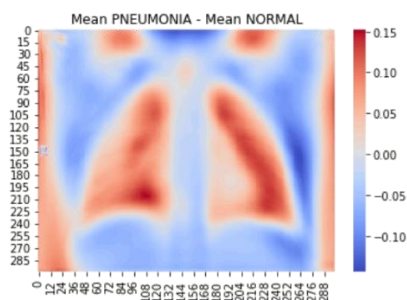


圖 5、兩組平均影像之差

分別將兩組影像各位置像素值計算變異數可得圖 6，由此圖可以觀察到兩組組內的差異主要是肺部輪廓以及軀幹輪廓，其中，最大變異數出現在正常組別，其值為 0.124，分布在接近頸部的地方。

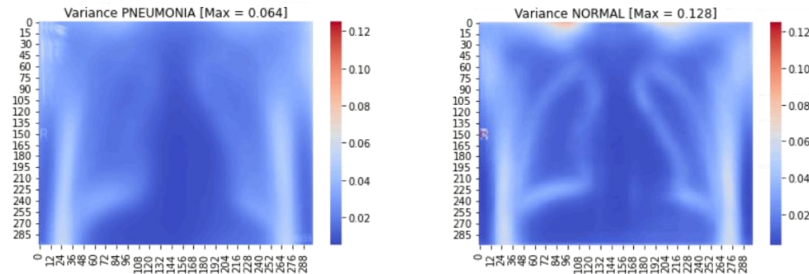


圖 6、肺炎與正常胸腔像素變異數影像

## 2. Project objective

給予胸腔 X 光影像，利用深度學習模型判斷為肺炎或為正常。

## 3. Methodology

由於使用的資料集資料量不多，因此採用 transfer learning，本作業實驗兩種方法，一為直接分類，另一為以物件辨識 (object detection) 找到胸腔位置後再分類。

### 3.1. 方法一

方法一參考 [2]。此論文使用的資料集與本作業相同，作者將資料集隨機以 0.88-0.12 比例分割為訓練與驗證資料集，在此改將資料集分割為訓練、驗證與測試資料集，將資料集依 0.8-0.1-0.1 比例分割，分別用於訓練、驗證與測試，實際分割資料量如表 1。驗證資料集用於模型訓練時評估模型表現以調整參數，測試資料集則用於確定模型後的最終表現評估。

論文的架構為將一部分訓練資料經過 data augmentation 之後輸入已使用 ImageNet 資料集預訓練的 ResNet18、DenseNet121、InceptionV3、Xception 與 MobileNetV2 進行 fine-tuning，得到個別的二元分類機率，接下來再用剩下的訓練資料訓練一個 weighted classifier，將前面所得的機率結果進行權重加總得到最終機率，機率大者為其歸類。

表 1、方法一之訓練、驗證與測試資料個數

	Train	Valid	Test
<b>Pneumonia</b>	3673	300	300
<b>Normal</b>	1083	250	250
<b>Total</b>	4756	550	550
<b>Percentage</b>	0.81	0.09	0.09

在此利用類似概念，架構為預訓練模型接上 weighted classifier，由於後者的概念與 fully connected layer 相似，因此這邊改用 fully connected layer 取代論文中的 weighted classifier，並且使用全部訓練資料直接訓練一個 end-to-end 預訓練網絡加上 fully connected layer 的模型，由於同時訓練多個 CNN 模型運算量十分龐大，因此，此作業中預訓練模型只選用論文中個別訓練 accuracy 前三高者（ResNet18、DenseNet121 與 Inception）。最後，論文使用 CAM 將訓練過程視覺化，由於此法必須在最後一層改接 GAP layer 才可得到結果，因此，此作業改使用 Grad-CAM。整個模型架構如圖 7。

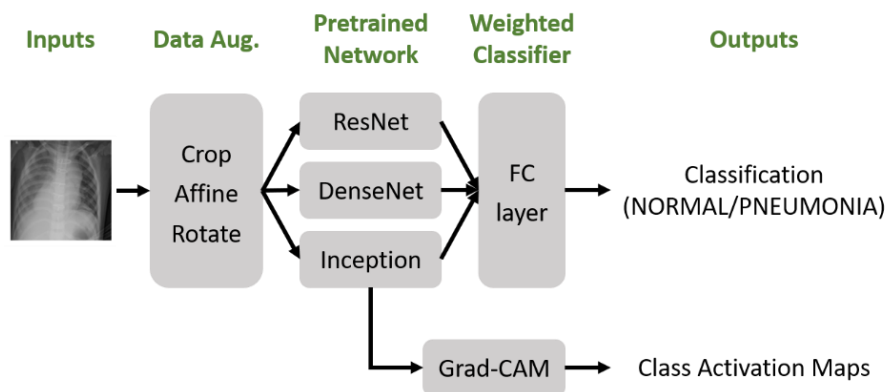


圖 7、方法一架構

簡單說明此作業使用的預訓練網絡：ResNet 主要由 convolution block 組成，並且引入 residual network 概念，在每個 block 中，第  $i$  層會與前面第  $i-2$  或  $i-3$  層進行元素級相加連接，做為下一層輸入；DenseNet 主要由 dense block 與 transition layer 組成，transition layer 由 convolutional layer 與 average pooling layer 構成，dense block 利用比 ResNet 更密集的连接機制，將每一層與前面所有層在 channel

維度相加連接，並做為下一層輸入；Inception V3 主要由 inception module 組成，利用稀疏連接結構，在同一層中融合不同 kernel 大小萃取特徵的結果。

### 3.2. 方法二

方法二參考 [3]，此論文作者利用已使用預訓練的 Faster R-CNN 辨識胸腔位置並且分類為感染肺炎或正常，不過其使用的資料集與本作業不同，因此嘗試將其方法套用於此作業使用的資料集。

Faster R-CNN 可分成三個部分，一為 backbone 用以取得 feature map，二為 RPN 網絡用以生成 region proposal（或稱 region of interest, RoI），即辨識出物件位置，最後是 RoI Pooling 與分類器，由前面所得的 feature map 與 RoI 計算出物件所屬的類別，架構如圖 8。

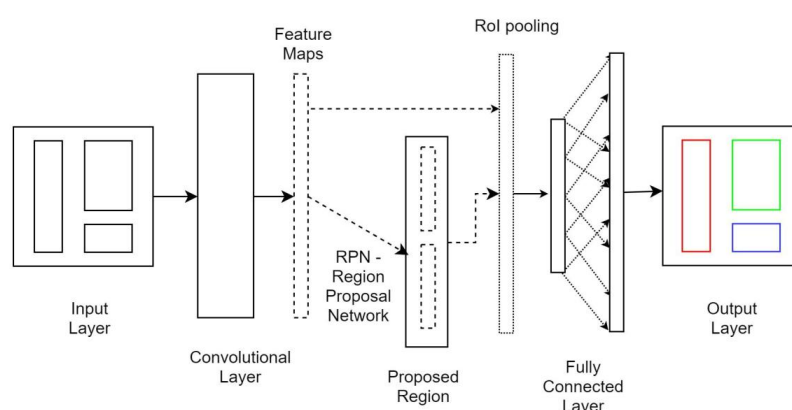


圖 8、Faster R-CNN 架構 [3]

由於肺炎的判定在臨床上是根據觀察肺部特徵，因此理論上模型也應根據肺部特徵進行分類。透過此方法可以確保模型辨識到胸腔位置，並且根據所框出胸腔位置內的特徵計算分類，避免模型得到正確的分類是因其他非肺部特徵。

論文中使用的資料集與本作業使用的資料集本身皆未標註肺部區域，故參考其方法，利用 Labellmg 軟體人工標註 200 張肺炎影像與 200 張正常胸腔影像，每張以方形框出胸腔所在區域，以鎖骨為上界，肋骨為左右與下界，將這 400 張標註資料以 0.8-0.2 比例分割用於訓練與驗證，剩餘資料則用於測試評估模型最終表現。模型的部分，採用在 COCO 資料集預訓練的 Fast R-CNN，修改 backbone 網絡。另外，資料預處理參考方法一進行 data augmentation。



## 4. Experiments

### 4.1. 方法一

參考論文 [2] 使用 TensorFlow 框架，本作業改使用 PyTorch 框架，預訓練模型與 data augmentation 方法使用 torchvision 套件 [4]，Grad-CAM 使用 pytorch-grad-cam 套件 [5]。訓練 30 個 epoch 並以 cross entropy 為 loss function，選擇在驗證資料集具有最低 loss 的參數模型用於測試。在此測試不同 data augmentation 方法（包括平移與旋轉）與不同 optimizer，而最後一層 weighted classifier 的部分測試改為單層 fully connected layer，將預訓練網絡輸出 128 維特徵向量後加總再通過兩層 fully connected layer 輸出。模型表現評估為將肺炎定為 positive，正常定為 negative，得到模型在測試資料集上 confusion matrix，再計算 accuracy 與 F1 score，最後以 Grad-CAM 視覺化三個預訓練網絡的訓練過程。

### 4.2. 方法二

參考論文 [3] 中使用 TensorFlow 框架，本作業改使用 PyTorch 框架，預訓練模型使用 torchvision 套件 [4]，data augmentation 方法使用 albumentations 套件 [6]。模型測試不同 backbone，包括 MobileNet 與 ResNet50，並且嘗試利用 data augmentation 進行資料預處理。

## 5. Results & Analysis

方法一的測試條件與結果如表 2。參考論文 [2] 中使用 SGD optimizer，因此首先採用相同 optimizer，儘管在表 2 中，比較相同條件下使用 Adam optimizer 的模型，在測試資料上 accuracy 與 F1 score 皆較高，但是觀察訓練過程（如圖 9 與圖 10），使用 SGD optimizer 訓練時 loss 一直無法下降，而使用 Adam optimizer 則可以成功收斂，因此後續實驗皆使用 Adam optimizer。

Data augmentation 測試使用平移與旋轉（Affine）與單純旋轉（Rotate），比較在測試資料的 accuracy 與 f1 score，使用 Affine 皆較高，因此判斷 Affine 較適合用此資料集訓練。不過，另外測試不進行 data augmentation 而直接取原始資料訓練，結果卻較使用 data augmentation 預處理後佳。

最後一層 weighted classifier 測試使用一層與兩層 fully connected layer 取代

可發現，不論是否有進行 data augmentation，後者在測試資料上 accuracy 與 f1 score 相較於前者皆較高，表示使用兩層 fully connected layer 分類效果更佳。

此方法測試模型分類效果最好為不將資料進行 data augmentation 預處理，使用 Adam optimizer 並且最後以兩層 fully connected layer 總和三個預訓練網絡的輸出，在測試資料上 accuracy 為 0.973、F1 score 為 0.975。

表 2、方法一測試條件與結果

	Data Augmentation	Optimizer	Weighted Classifier	Acc	F1
1	Affine(deg=20, translate=(0.15, 0.2))	SGD	1 FC (6)	0.965	0.970
2	Affine(deg=20, translate=(0.15, 0.2))	Adam	1 FC (6)	0.960	0.963
3	Rotate(deg=30)	Adam	1 FC (6)	0.956	0.960
4	No	Adam	1 FC (6)	0.967	0.970
5	Affine(deg=20, translate=(0.15, 0.2))	Adam	2 FC (128, 64)	0.962	0.966
6	No	Adam	2 FC (128, 64)	<b>0.973</b>	<b>0.975</b>

Acc = accuracy, F1 = F1 score

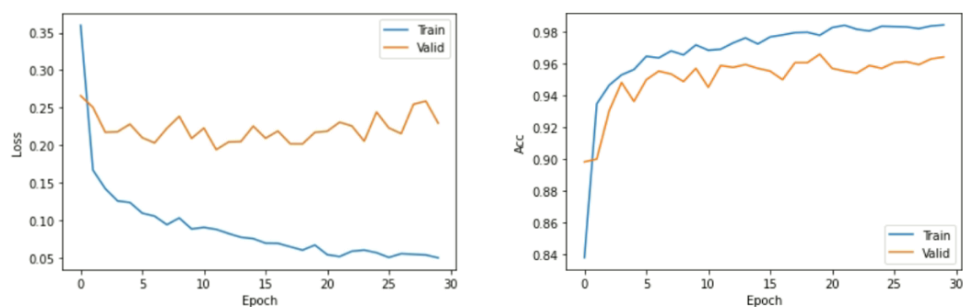


圖 9、使用 SGD optimizer 訓練過程 loss 與 accuracy 變化



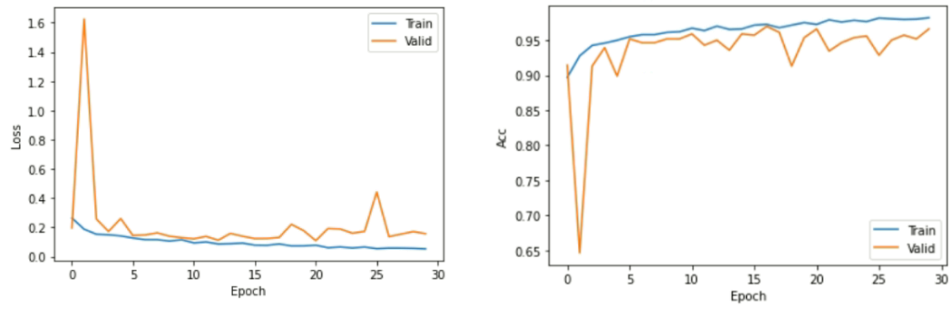


圖 10、使用 Adam optimizer 訓練過程 loss 與 accuracy 變化

接下來以 Class Activation Map (CAM) 觀察三個預訓練網絡的訓練過程，選用表 2 中第 6 組條件的模型，利用 Grad-CAM 視覺化如圖 11。

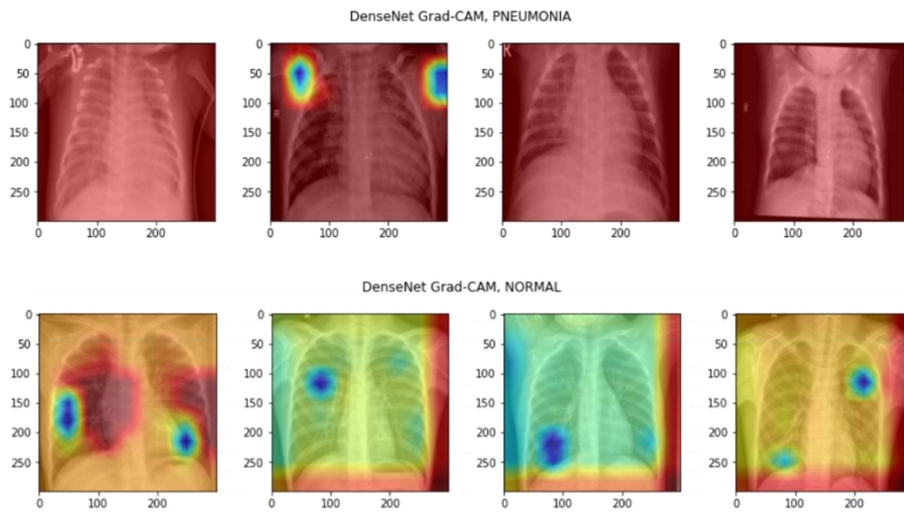


圖 11-a、DensNet121 之 Grad-CAM 結果

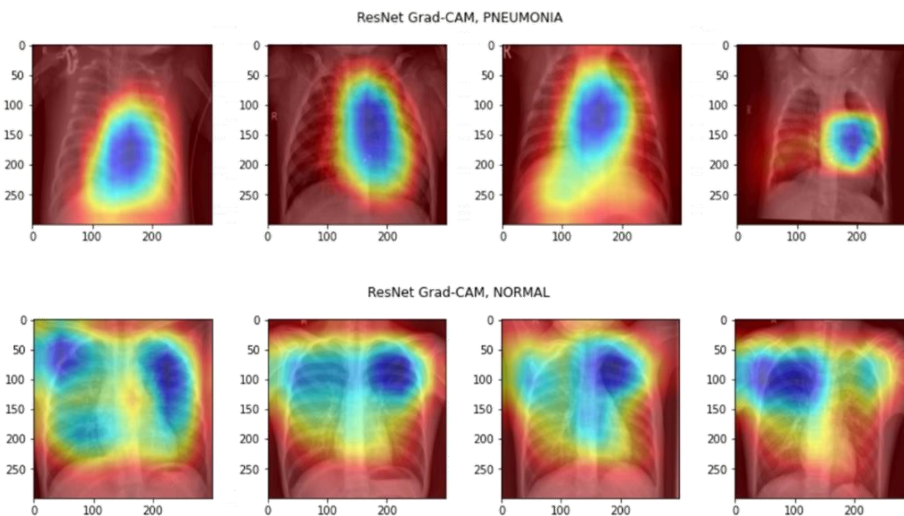


圖 11-b、ResNet18 之 Grad-CAM 結果

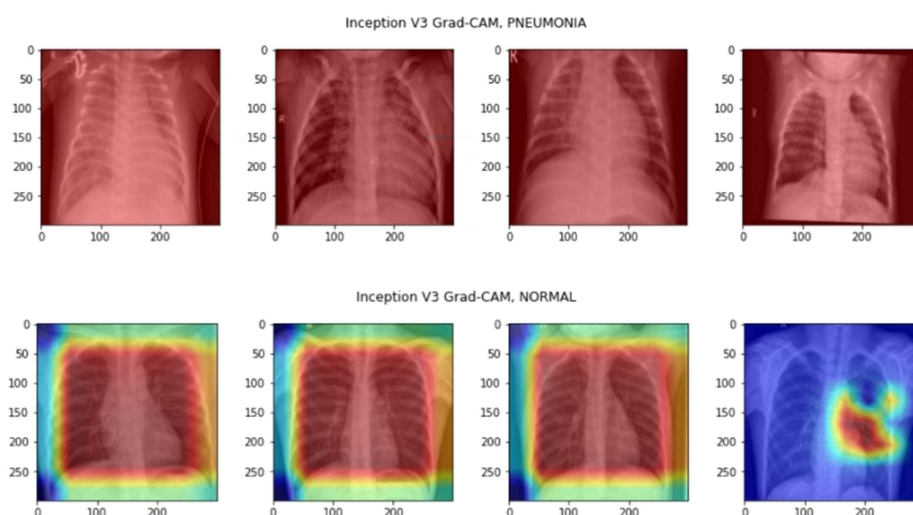


圖 11-c、Inception V3 之 Grad-CAM 結果

由圖 11 可發現雖然經過測試顯示模型成功分類高達 97% 測試資料，然而模型大多辨識到非肺部的特徵。因此，之後決定測試使用物件辨識的方法，讓模型辨識胸腔位置再進行分類，以此避免模型根據非預期的特徵而得到正確分類。

參考論文 [3] 使用 Faster R-CNN 進行物件辨識與分類，測試條件與結果如表 3。可以發現 backbone 使用 MobileNet 在測試資料上具有較高的 accuracy 與 F1 score。而與方法一的結果相似，使用 data augmentation 之後測試的結果反而變差。表 3 相較於表 2 數值較低，然而由於訓練資料不同，因此無法直接進行比較。

表 3、方法二測試條件與結果

	Data Augmentation	Backbone	Acc	F1
1	No	Resnet50	0.936	0.956
2	No	MobileNet	<b>0.941</b>	<b>0.960</b>
3	Affine(deg=20, translate=(0.15, 0.2))	Resnet50	0.928	0.951
4	Affine(deg=20, translate=(0.15, 0.2))	MobileNet	0.933	0.954

Acc = accuracy, F1 = F1 score

選取表 3 中第 2 組條件的模型對測試資料進行預測並且畫出其辨識情形，如圖 12，可發現模型正確辨識胸腔的位置並且給予其分類，符合預期結果。若人工標註用於訓練的資料量增加，或許此方法的準確度可以再提高。

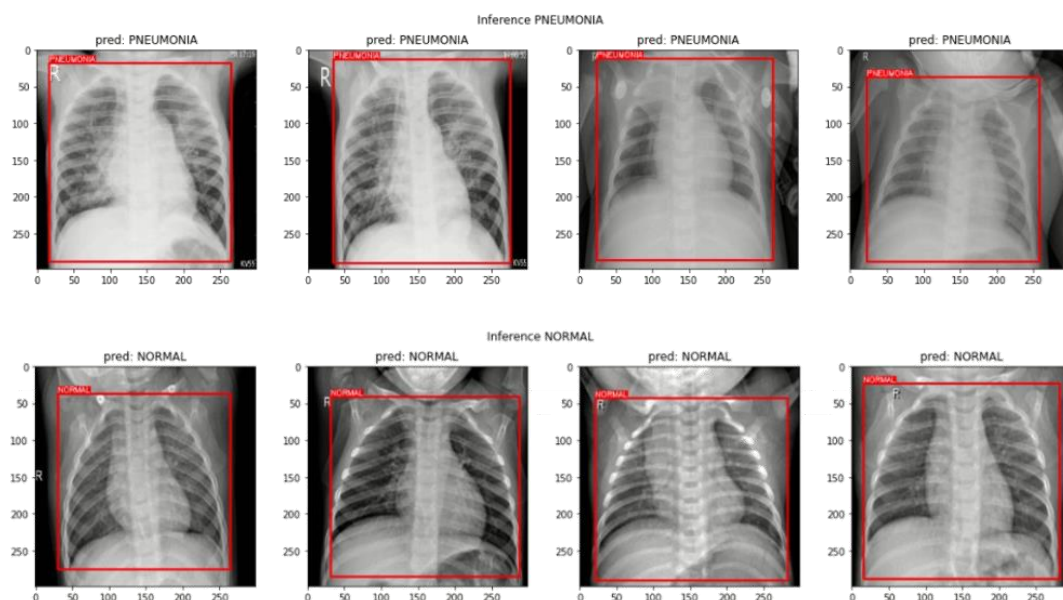


圖 12、Faster R-CNN 辨識與分類預測

## 6. Reference

- [1] D. Kermany, “Labeled Optical Coherence Tomography (OCT) and Chest X-ray Images for Classification,” *Mendeley Data*, 06-Jan-2018. [Online]. Available: <https://data.mendeley.com/datasets/rscbjbr9sj/2>.
- [2] M. F. Hashmi, S. Katiyar, A. G. Keskar, N. D. Bokde, and Z. W. Geem, “Efficient Pneumonia Detection in Chest Xray Images Using Deep Transfer Learning,” *Diagnostics*, vol. 10, no. 6, p. 417, 2020.
- [3] A. Ismail, T. Rahmat, and S. Aliman, “Chest X-ray Image Classification Using Faster R-CNN,” *Malaysian Journal of Computing*, vol. 4, no. 1, p. 225, 2019.
- [4] “Torchvision.” [Online]. Available: <https://pytorch.org/vision/0.12/>.
- [5] “Class Activation Map Methods Implemented in Pytorch.” [Online]. Available: <https://github.com/jacobgil/pytorch-grad-cam>.
- [6] “Bounding Boxes Augmentation for Object Detection.” [Online]. Available: [https://albumentations.ai/docs/getting\\_started/bounding\\_boxes\\_augmentation/](https://albumentations.ai/docs/getting_started/bounding_boxes_augmentation/).