

國立雲林科技大學

資訊管理所

機器學習專案二報告

M11123050 李京樺

M11123036 梁滄愷

M11123038 曾治程

M11123016 陳秉言

指導教授：許中川教授

中華民國 110 年 5 月

# 摘要

本次實驗使用機器學習方式將氣管做切割，透過自動化影像切割能節省時間與人力成本，提高效率和準確性。本實驗能夠從 X 光片中準確的提取氣管所在的區域、大小、形狀、位置等訊息。有助於醫護人員能夠觀察氣管內部構造。

本實驗使用 Unet、Unet++，同時比較各模型在不同超參數下的績效，並分別透過 Intersection over Union (IOU) 衡量指標評估各個模型對於數值的績效，並比較其結果。結果顯示兩個模型在 learning rate 和 batch\_size 都是越低績效越好，而在 epoch 比較上是 Unet 的績效較好。

**關鍵字：**Unet；Unet++；影像切割；深度學習

# 一、緒論

## 1.1 動機

本次實驗使用機器學習方式將氣管做自動化切割，氣管是呼吸系統中最為重要的器官，與氣管有關的疾病包含氣喘、塵肺症、間質性肺病、肺結炎與其他慢性呼吸道疾病 ( Momtazmanesh et.al.,2023)。透過對氣管影像進行切割，可以提高對於疾病診斷的精確度與治療的效果，透過使用機器學習的技術進行自動化的氣管切割也能節省時間與人力成本，提高效率和準確性。

## 1.2 目的

透過本實驗能夠從X光片中準確的提取氣管所在的區域、大小、形狀、位置等訊息。醫學人員可以觀察和分析氣管內部的結構，並能進一步的對氣管疾病進行診斷與治療。同時也能減少讓醫學人員使用影像軟體手動標註氣管位置，所產生大量時間和人力成本。

# 二、方法

本實驗所使用的方法分為四個階段，第一階段為匯入資料集，將五個 Fold 檔案訓練、驗證與測試 X 光影像，與遮罩資料分別匯入以進行後續的實驗。第二階段為做資料前置處理，將匯入的影像轉為灰階，並使用resize技術將圖形尺寸轉為256\*256，以方便模型的運算。第三階段為做資料強化，透過將影像進行縮放、旋轉、水平平移、垂直平移等處理，以創造出更多的圖片讓模型學習，彌補訓練資料不足的問題。第四階段構建出模型使用Unet、Unet ++，分別進行模型超參數的設定並進行訓練，訓練完的模型使用IOU指標評估各模型的績效。更詳細的內容如實驗步驟所說明。

## 三、實驗

### 3.1 資料集

本實驗所使用的資料集是 ETT\_V3，裡面分成 5 個資料夾，其中包含，訓練集 (train)、訓練遮罩 (trainannot) 由圖 1 顯示，驗證集 (val)、驗證遮罩 (valannot) 由圖 2 顯示、測試集 (test) 以及測試遮罩 (testannot) 由圖 3 顯示。資料筆數分別為訓練集 287 筆、訓練遮罩 287 筆、驗證集 47 筆、驗證遮罩 47 筆、測試集 47 筆以及測試遮罩 47 筆。由於資料筆數過多無法全部呈現。



圖 1 訓練集和訓練遮罩



圖 2 訓練集和訓練遮罩

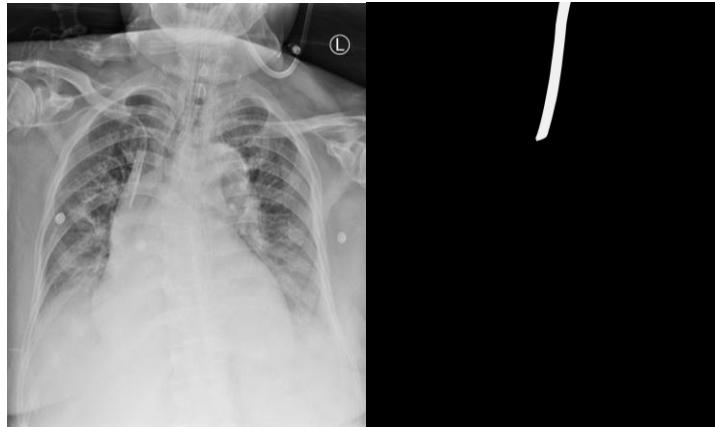


圖 3 測試集和測試遮罩

## 3.2 前置處理

### 3.2.1 讀取資料夾內檔案

透過讀取圖片資料夾檔案所在的路徑，並定義 `resizeimg` 函數，將檔案路徑拆解成串列型態的字串，並將串列進行反轉，取得第一個字串為路徑內圖片的名稱。

### 3.2.2 導入相關套件

導入 `numpy`、`os`、`PIL`，`tensorflow`、`tensorflow_datasets`、`cv2` 等套件，並匯入 `tensorflow.keras.preprocessing.image` 裡的 `load_img` 與 `tensorflow.keras.layers` 裡的 `Input`、`Dense`、`Dropout`、`Activation`、`Flatten`、`Conv2D`、`MaxPooling2D`、`Conv2DTranspose`、`concatenate` 與 `Model`。

### 3.2.3 圖片處理

透過使用 Keras 深度學習框架中的 `load_img` 函數，用於將圖片轉為 Keras 模型可以接受的格式。設定 `targer_size` 將圖片尺寸轉換為 `256*256`。將每張圖片資料除以 255 進行正規化，並使用 OpenCV 套件將圖片從 RGB 轉為灰階。

### 3.2.4 資料強化

透過使用 keras 的 ImageDataGenerator 做圖片的強化，有助於模型的訓練和收斂，旋轉 `rotation_range = 10`，把圖片隨機旋轉角度於10度到 -10度之間、平移 `width_shift_range = 0.1`及 `height_shift_range= 0.1`，把圖片水平及垂直隨機平移 10% 的範圍內，可增加模型的泛化能力，最後利用 `horizontal_flip` 把圖片進行水平翻轉。

### 3.3實驗設計

本實驗共建構兩個模型，分別為 Unet 與 Unet++ 模型。將資料強化過的圖片資料放入至模型當中。兩個模型皆使用 Adam 做為優化器，損失函數使用 `binary_crossentropy`。利用 early stopping 技術找出 epochs 值，設定patience 值為 3，當模型連續 3 次內驗證資料的損失函數皆無改善就停止訓練，以防止過度擬合的產生。再透過修改調整 learning rate 和 batch\_size 等超參數去比較模型之間 IOU 及 驗證資料的損失函數( `binary_crossentropy` )。

### 3.4實驗結果

#### 3.4.1 比較 Unet 與 Unet ++ Learning rate

以下表1及表2代表 Unet 和 Unet++不同模型之下所設定不同的 Learning rate 比較IOU 及驗證資料的損失函數。

表 1

在 Unet 模型下調整不同的 Learning rate 之 IOU 及驗證資料的損失函數。

Learning rate	IOU	valloss
0.01	0.4939	0.0568
0.001	0.5632	0.0195
0.0001	0.7709	0.0100

表 2

在 Unet++ 模型下調整不同的 Learning rate 之 IOU 及驗證資料的損失函數。

Learning rate	IOU	valloss
0.01	0.4942	0.0565
0.001	0.5687	0.0214
0.0001	0.7710	0.0117

### 3.4.2 比較Unet 與 Unet ++ batch\_size

以下表3 及表4 代表Unet 和 Unet++不同模型之下所設定不同的 batch\_size 比較 IOU 及驗證資料的損失函數。

表 3

在 Unet 模型下調整不同的 batch\_size 之 IOU 及驗證資料的損失函數。

batch_size	IOU	valloss
2	0.7709	0.0100
4	0.7041	0.0147
8	0.6775	0.0166

表 4

在 Unet ++ 模型下調整不同的 batch\_size 之 IOU 及驗證資料的損失函數。

batch_size	IOU	valloss
2	0.7710	0.0117
4	0.7686	0.0129
8	0.4940	0.0482

### 3.4.3 比較 Unet 與 Unet ++ 在不同 epoch 下的 IOU 及驗證資料的損失函數

以下圖 4 和圖 5 代表 Unet 和 Unet++ 不同模型之下不同的 epoch 比較 IOU 及驗證資料的損失函數。

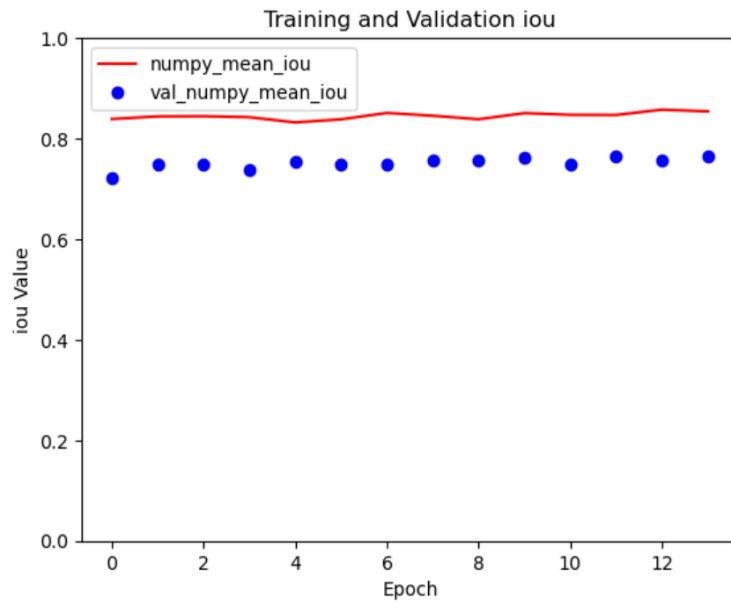


圖 4 Unet 在不同 epoch 下的 IOU 及驗證資料的損失函數

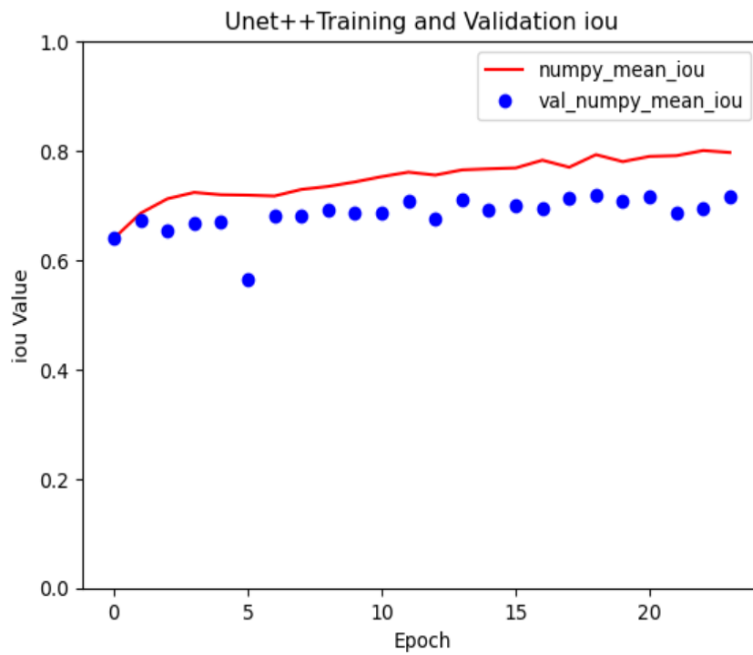


圖 5 Unet ++ 在不同 epoch 下的 IOU 及驗證資料的損失函數



### 3.4.4 各 Fold之間 Unet 與 Unet++ 的視覺化預測結果

以下圖 6 到圖 15 之間是各 Fold 之間 Unet 與 Unet ++ 的視覺化預測結果。

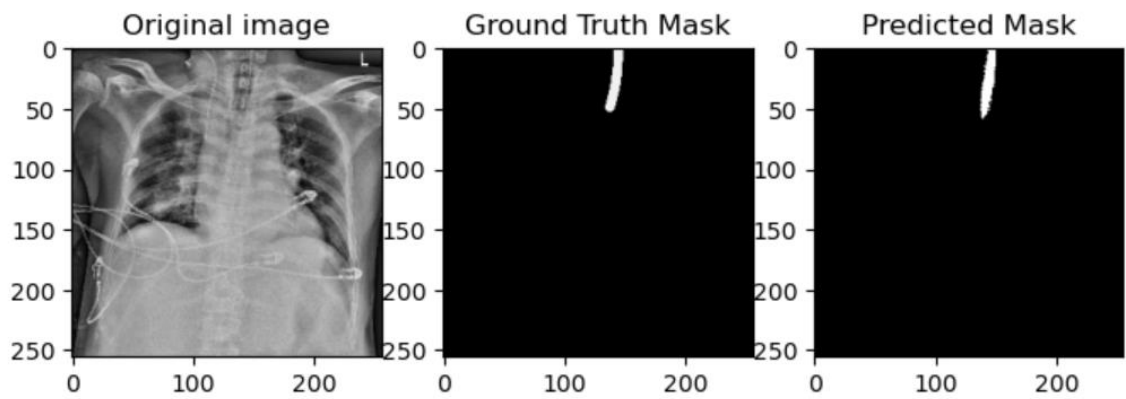


圖 6 Unet Fold1 視覺化預測結果

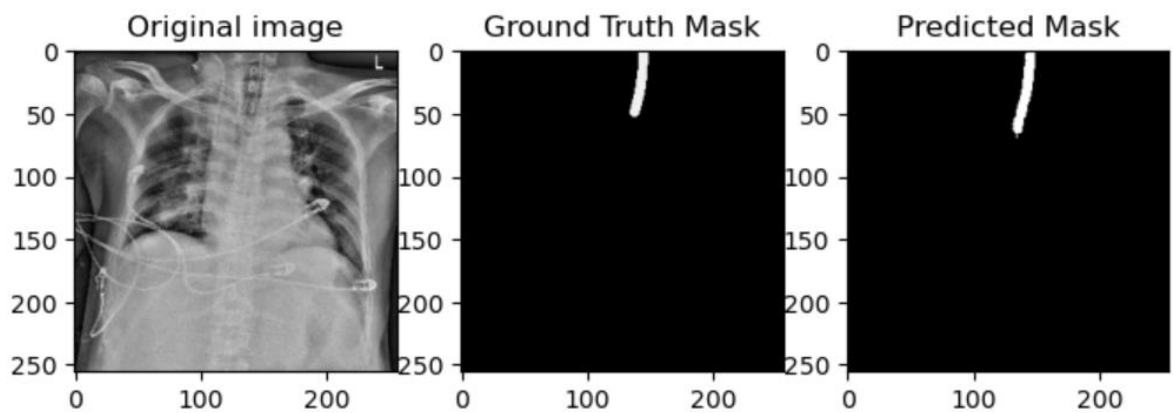


圖 7 Unet ++ Fold1 視覺化預測結果

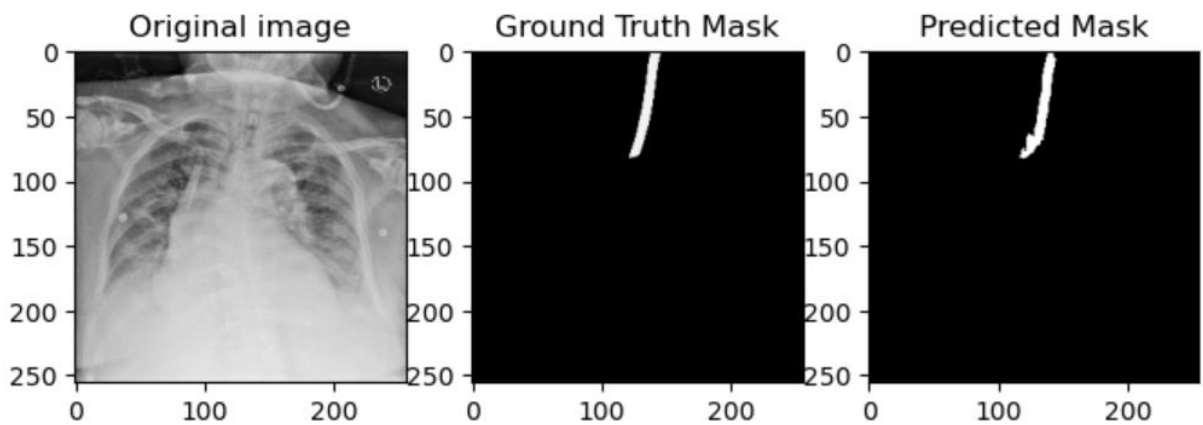


圖 8 Unet Fold2 視覺化預測結果

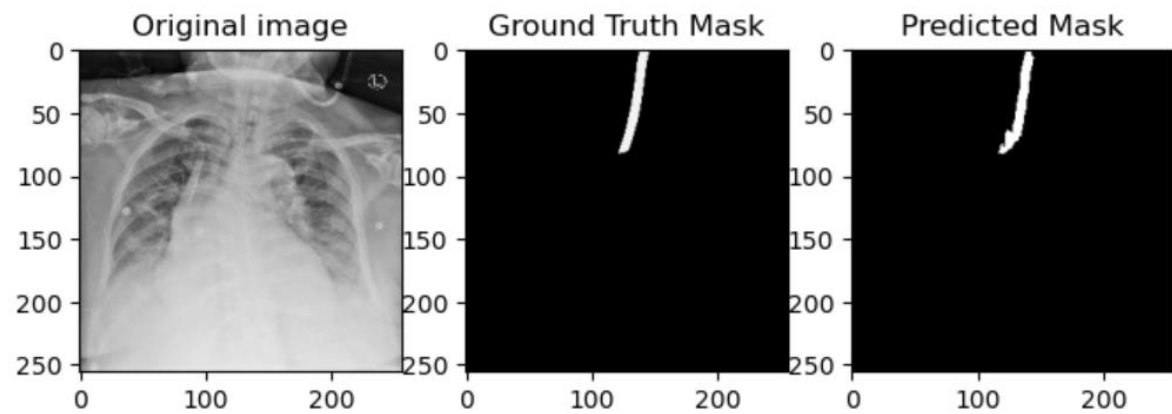


圖 9 Unet ++ Fold2 視覺化預測結果

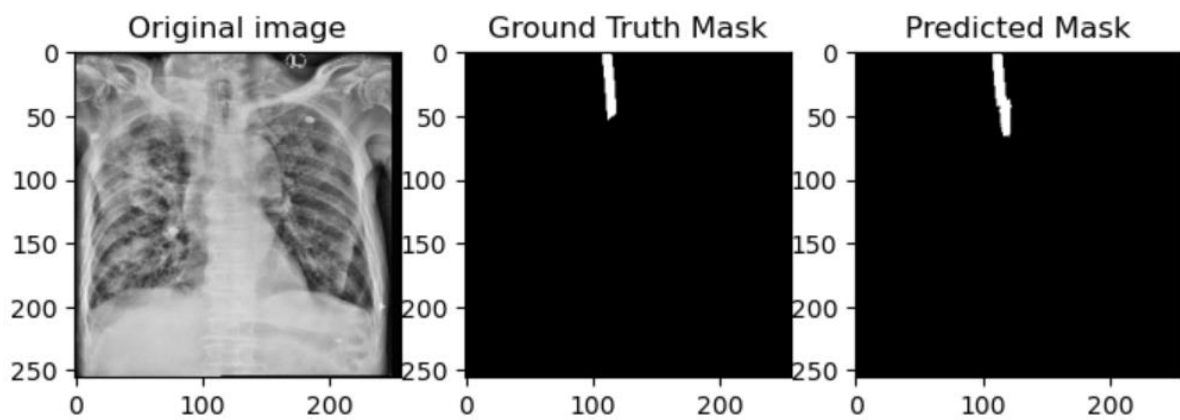


圖 10 Unet Fold3 視覺化預測結果

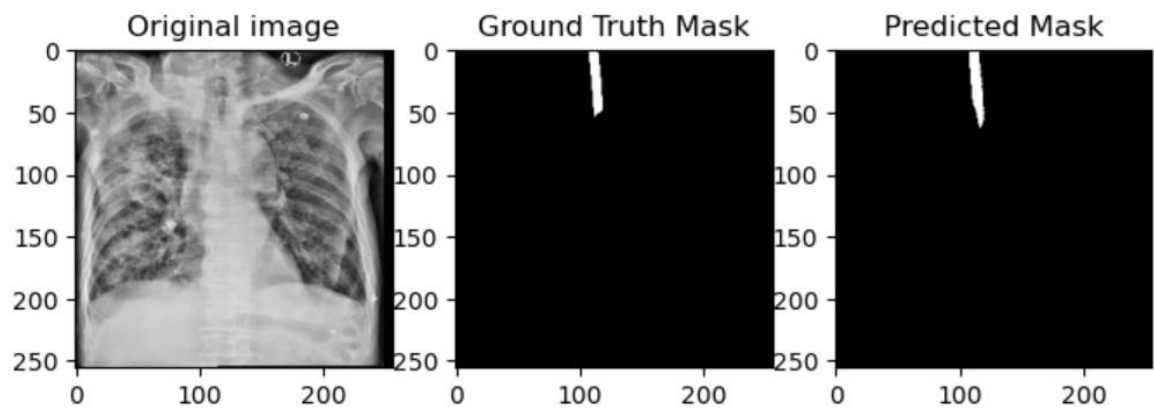


圖 11 Unet ++ Fold3 視覺化預測結果

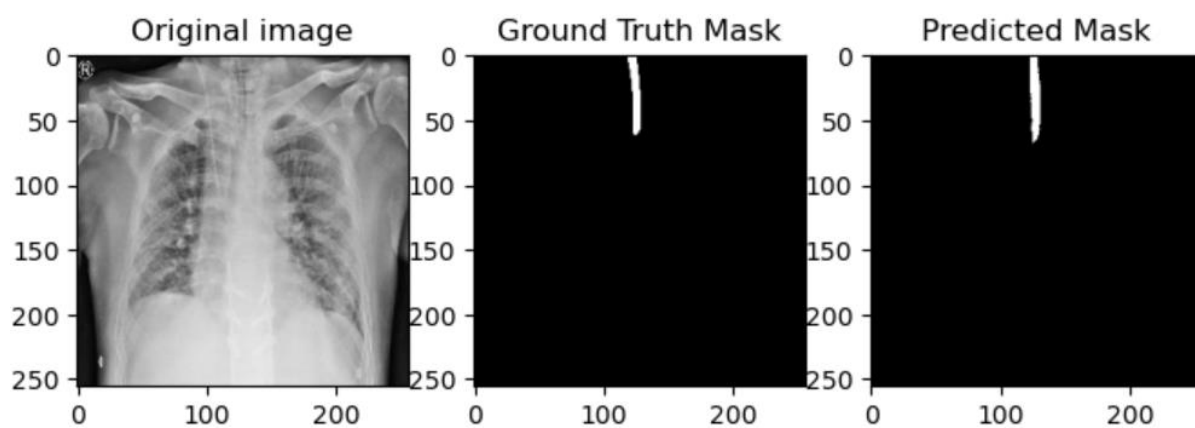


圖 12 Unet Fold4 視覺化預測結果

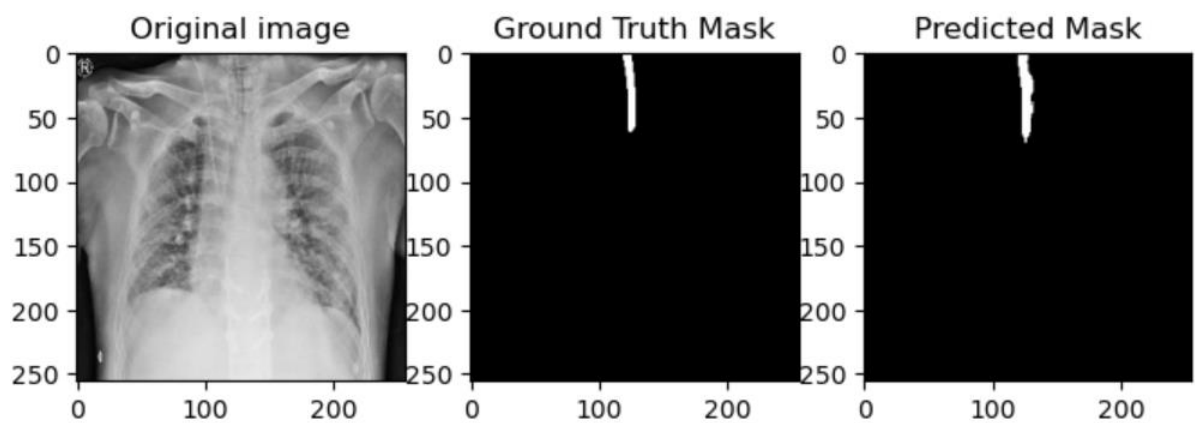


圖 13 Unet ++ Fold4 視覺化預測結果

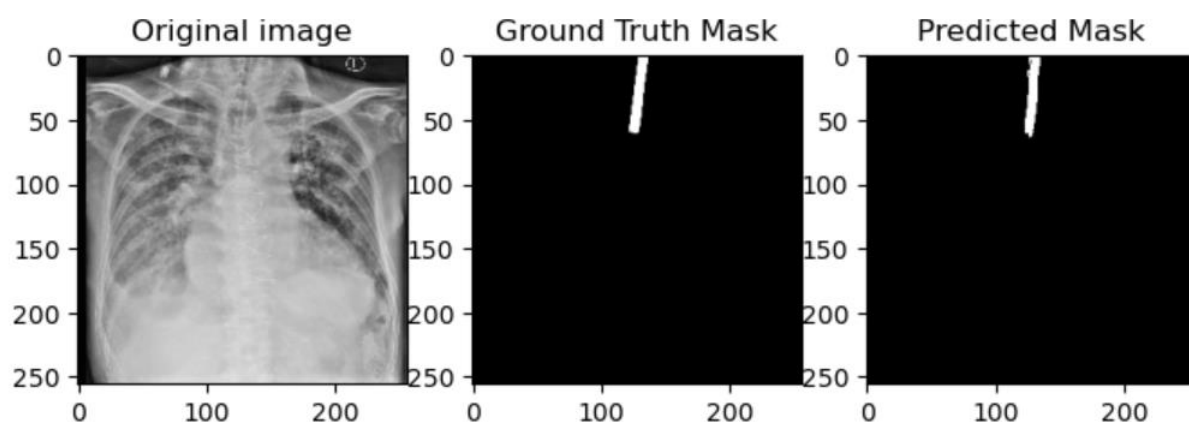


圖 14 Unet Fold5 視覺化預測結果

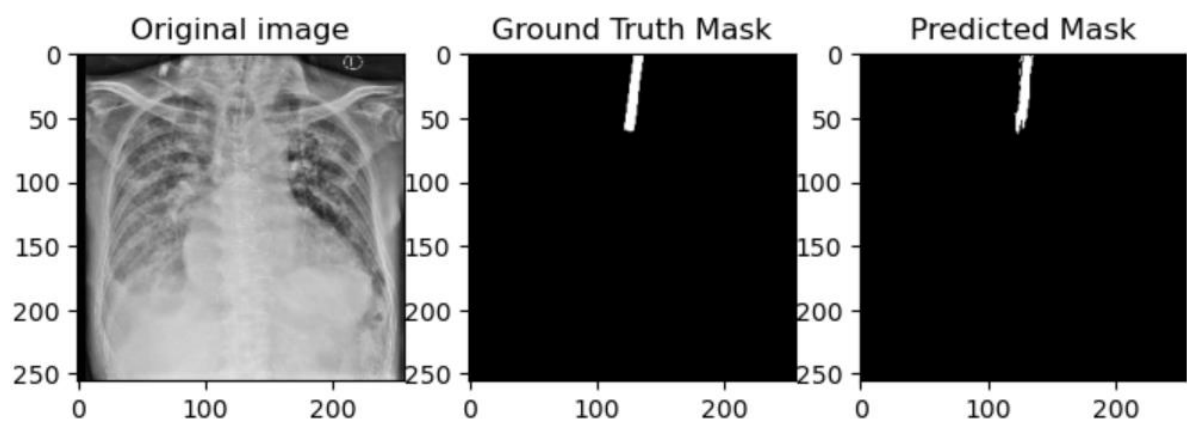


圖 15 Unet ++ Fold5 視覺化預測結果

### 3.4.5 各Fold間的績效比較

以下表5為Unet 的各個Fold IOU 的績效與平均，表6為Unet ++ 的各個Fold IOU 的績效與平均。

表 5

各Fold間 Unet 的績效比較

	預測資料	測試資料
fold1	0.8211	0.7210
fold2	0.8172	0.7342
fold3	0.7882	0.7362
fold4	0.7799	0.7272
fold5	0.7492	0.6766
Average	0.7911	0.7186

表 6

各 Fold 間 Unet ++ 的績效比較

	預測資料	測試資料
fold1	0.7970	0.7169
fold2	0.8017	0.7143
fold3	0.7752	0.7024
fold4	0.7542	0.7047
fold5	0.7249	0.6467
Average	0.7706	0.6927

## 四、結論

在本實驗中，先使用了圖像增強的方法彌補一開始訓練資料不足與避免訓練產生過度擬合的問題，再把圖片放進 Unet 和 Unet ++ 兩個模型進行圖片遮罩的預測，訓練時透過設定不同的 learning rate 和 batch\_size 下來比較模型之間 IOU 及驗證資料的損失函數。從實驗結果來看，兩個模型在 learning rate 和 batch\_size 上都是越小績效越好，而在比較兩個模型的 epoch 績效上，是 Unet 比 Unet ++ 有更好的績效。

## 參考文獻

Momtazmanesh, S., Moghaddam, S. S., Ghamari, S. H., Rad, E. M., Rezaei, N., Shobeiri, P., & Ibitoye, S. E. (2023). Global burden of chronic respiratory diseases and risk factors, 1990–2019: an update from the Global Burden of Disease Study 2019. *EClinicalMedicine*.