

# Guide

## 檔案說明

### 1. preprocessing 資料夾

檔案名稱	說明
data_extracting.ipynb	從原始資料集中擷取所需要使用的資料，生成 annotation.csv 及 coords.pickle。
data_generator.ipynb	生成 axial、coronal 及 sagittal 的輸入影像 original images 及 final_annotation.csv(labels)。
multiview_generator.ipynb	將 axial、coronal、sagittal planes 連接起來。
texture_concatenation.ipynb	將紋理特徵跟 original images 連接起來。
data_augmentation.ipynb	執行資料增強(旋轉、翻轉)。
cross_val.ipynb	將資料打亂後分成十等份。
util.py	自製函式庫。
data_generator-plot.ipynb	畫圖用。
data_visualization.ipynb	畫圖用。
data_visualization2.ipynb	畫圖用。

### 2. texture feature generation 資料夾

檔案名稱	說明
generator.m	產生紋理特徵的影像。
other	計算紋理特徵。

### 3. classification 資料夾

檔案名稱	說明
train.ipynb	可對選定的資料做 10-fold 交叉驗證。
train_run_all.ipynb	對每一種紋理特徵做 10-fold 交叉驗證。
MRSKNet.py	train.ipynb 所使用的模型。
MRSKNet2c.py	Train_run_all.ipynb 所使用的模型。
ResNet.py	BasicResNet。
SKNet.py	BasicSKNet。
XXXNet.py	模型架構。
util.py	自製函式庫。

## 資料處理

1. 解壓縮 material 資料夾中的 LIDC.zip。
2. 開啟 preprocessing 資料夾，開啟 jupyter notebook，然後執行 data\_extracting.ipynb，依照 list3.2.csv(LIDC 結節大小報告)從 LIDC-IDRI 原始資料中篩選出肺結節，執行完畢會產生 annotation.csv 及 coords.pickle。  
[1]: lidc\_folder 為步驟 1 解壓縮的資料夾路徑。  
[1]: data\_folder 為輸出的資料夾路徑。
3. 執行 data\_generator.ipynb，對資料做預處理，會得到每顆肺結節的 axial、coronal 及 sagittal 平面，以及記錄 label 的 final\_annotation.csv。  
[1]: lidc\_folder 及 data\_folder 同步步驟 2。  
[1]: images\_path 為輸出的資料夾路徑。
4. 使用 MATLAB 執行 generator.m，生成各種不同的紋理特徵。  
第 2 行: CoreNum 可調整使用 CPU 的數量來平行加速。  
第 8 行: images\_path 為用來擷取特徵的影像，original images 的路徑。  
第 26 行: N=1 為使用 3x3 的 sliding window，N=2 為 5x5。  
第 76, 78, ~86, 88 行: 輸出紋理特徵影像的路徑。
5. 回到 jupyter notebook 執行 multiview\_generator.ipynb，將 axial、coronal、sagittal 三種平面連接在一個新的維度上。  
[1]: data\_folder 為輸入的資料夾路徑。  
[4]: output\_path 為輸出的資料夾路徑。
6. 執行 texture\_concatenation.ipynb，將 original images 跟每一種紋理特徵在 channel 維度做連接。  
[1]: data\_folder 為輸入的資料夾路徑。  
[3]: output\_path 為輸出的資料夾路徑。
7. 執行 data\_augmentation.ipynb，將資料做資料增強。  
[1]: data\_folder 為輸入的資料夾路徑。  
[3]: augimgs\_folder 為輸出的資料夾路徑。
8. 執行 corss\_val.ipynb，將全部資料隨機分成 10 等份，生成 tenFold.pickle，裡面記錄每一等份的肺結節 id。  
[1]: data\_folder 為輸入與輸出的資料夾路徑。  
[4]: df 為讀取 final\_annotation.csv。  
[4]: imgs\_path 為 multi\_view 底下 original images 的路徑。

## 訓練

1. 打開 classification 資料夾。
2. 執行 train.ipynb，訓練時會做 10-fold 交叉驗證，最後生成一個 txt 檔案生成訓練結果，以及畫出 10-fold 交叉驗證的 ROC 曲線。

- [1]: `data_folder` 為輸入的資料夾路徑。
  - [4]: `dataset` 為要使用的資料集，可以使用 `original`、`ENT`、`HOM`、`GLN`、`RLN`、`RP`、`SRE`、`COA`。
  - [4]: `df` 為讀取 `final_annotation.csv`。
  - [4]: `tenFold` 為讀取 `tenFold.pickle`。
  - [4]: 第 59 行選擇 `train()` 裡面的 `model`，可以使用 `MRSKNet`、`BasicResNet`、`BasicSKNet`。
  - [5]: 將 10-fold 交叉驗證結果寫入 `txt`，可以自己填寫檔名。
  - [6]: 可以看每個 `fold` 訓練的 `loss`。
  - [9]: 畫出 10-fold 的 ROC 曲線並且儲存。
3. 執行 `train_run_all.ipynb`，可以對每一種加入紋理特徵的影像做訓練，並且生成 `txt` 檔案記錄每一種加入紋理特徵的實驗訓練後的最終結果，以及畫出每一種加入紋理特徵的實驗訓練後的平均 ROC 曲線。
- [1]: `data_folder` 為輸入的資料夾路徑。
  - [2]: 第 4 行開啟要寫入訓練結果的 `txt` 檔案。
  - [6]: 畫出每一種加入紋理特徵的實驗訓練後的平均 ROC 曲線。