

# Image Processing Final Project

## P76084091 李宗樺

### A. Introduction:

從脊柱X-ray圖像可以看出脊椎側彎的程度。正確的找出每一塊脊椎骨的位置對於判讀脊椎側彎是很重要的。

作業要求自動從脊椎X-ray影像分割出脊椎區域，特別要求對每一塊脊椎骨計算DICE。

### B. Methods:

常見醫療影像處理都是用U-Net，我這次使用U-Net 的變形——Dense-Unet 作為我的基礎模型，Dense-Unet 結合 densenet (Fig 1) 的 dense block，在convolution 後會在加上之前low-level的資訊。會選擇使用這個模型的原因是因為我在其他影像分類的問題上有用到，而dense net 的特色在於密集連接的機制：每個layer 都會加上之前layer的輸出。

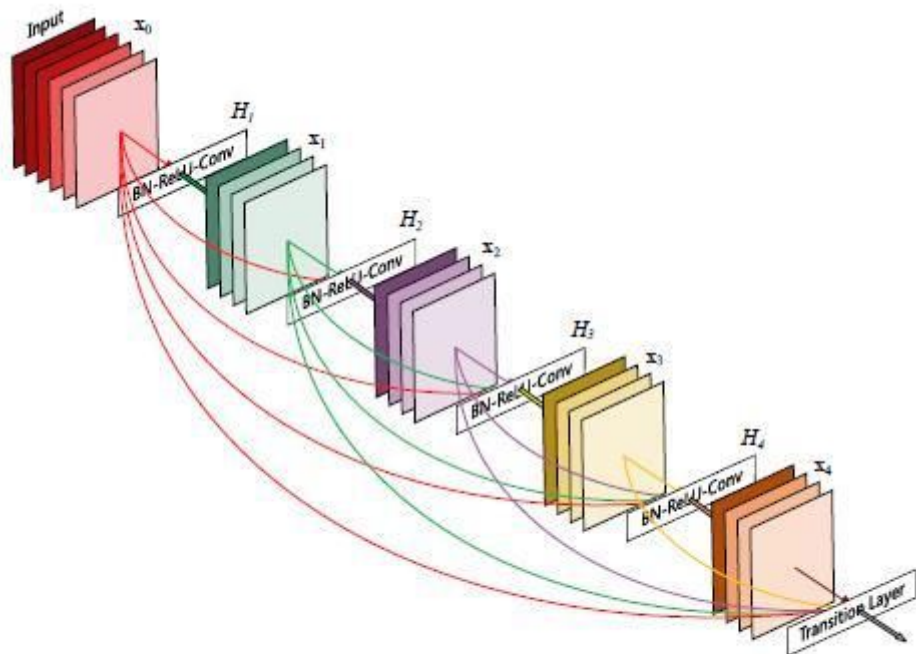


Fig 1

在原論文中有提到這樣的機制使得feature 和 gradient 更容易傳遞。

另外U-Net特色在於連結encoder 和 decoder 之間會做連接，可以充分使用low level的資訊，對於醫療影像有很大的幫助，而我認為加入dense block （Fig 3）可以更進一步強化這個特色。

其實已經有不少論文都是使用Dense-Unet的架構，本篇實做方法參考論文 Fully Dense UNet for 2D Sparse Photoacoustic Tomography Artifact Removal 以及 H-DnseUnet:Hybrid Densely Connected UNet for Liver and Tumor Segmentation from CT Volumes （Fig 2）。

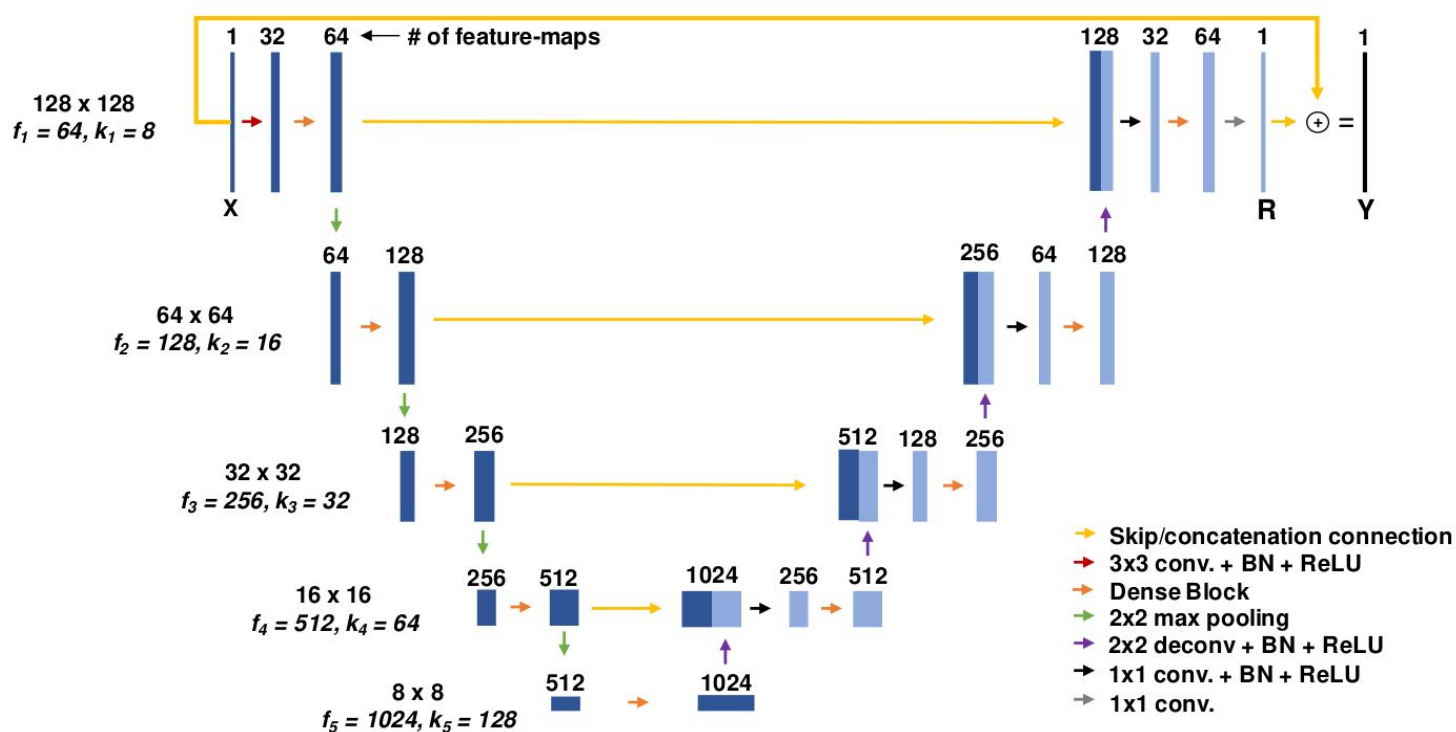


Fig 2 Demse Unet

透過參數growth-rate控制encoder dense block 的output channel。

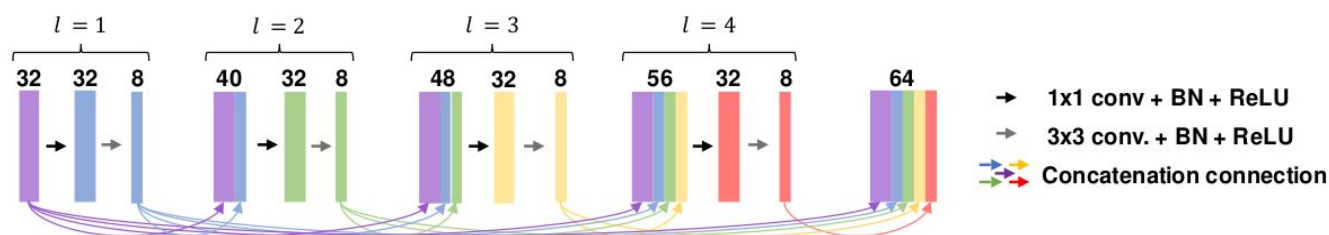


Fig3 Dense Block

## Data Preprocessing:

將所有影像Normalize成0~1的數值，原圖大小1200\*500 在 down sampling 會無法整除，所以同一個Layer Encoder 和Decoder 維度會不相同，為解決這問題我會將原圖切成 1200\*400（左右減少50個 pixel）。

## Loss:

原本只使用dice 作為loss function，發現雖然總dice 很高，但是個別脊椎的dice 並不理想，會有很多雜訊，因此結合focal loss（Fig 4），focal loss是針對easy example進行down-weighting，目的是在訓練過程中能盡量去訓練hard example。

$$FL(p_t) = -\alpha(1 - p_t)^r \log(p_t)$$

Fig 4 Focal Loss

## Group Normalization:

常見都是使用Batch Normalization，但是在Batch size 很小時，會嚴重影響model performance，因此我改用大神Kaiming He 在2018 年提出的group normalization(GN)，在該paper的實驗中 GN幾乎不受batch size 影響。

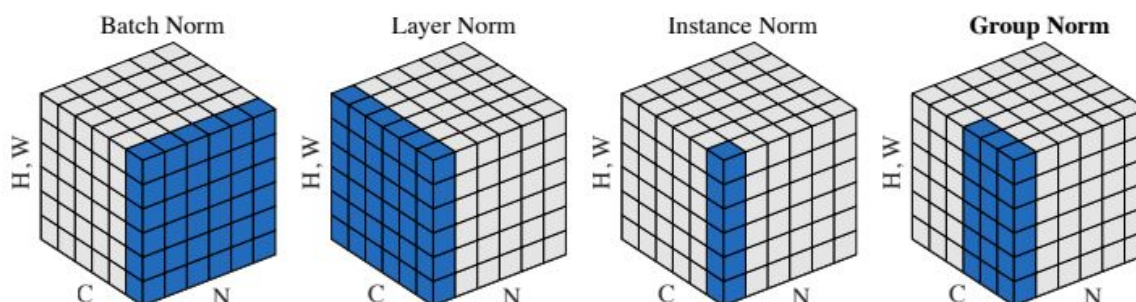


Fig 5 Compare Normalized

## Fold-Train:

使用3-fold cross validation 訓練模型。

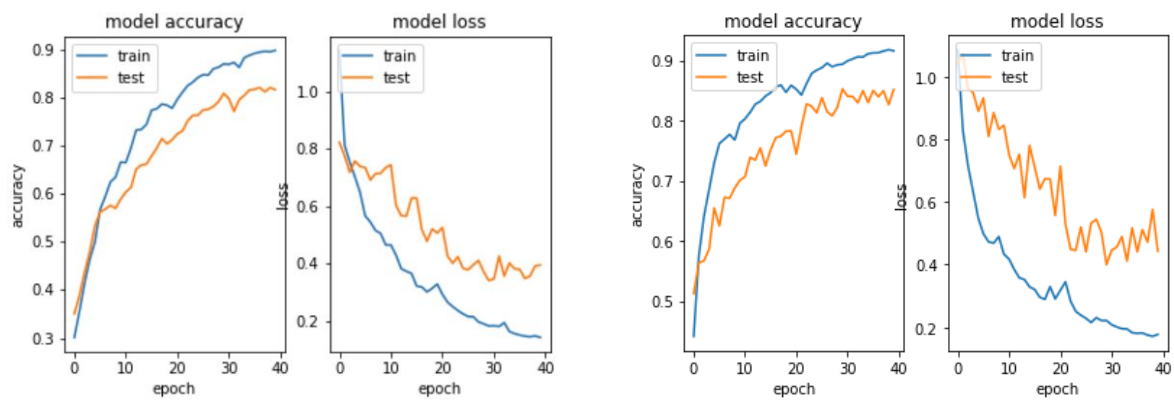
## 計算個別脊椎骨Dice:

在當初訓練model 時沒有將個別脊椎骨分開預測，模型只能計算整張圖的Dice結果，所以只好透過後處理的方式。先將ground truth 和 predict 二值化並做connect component，再計算個別dice。

## C. Results:

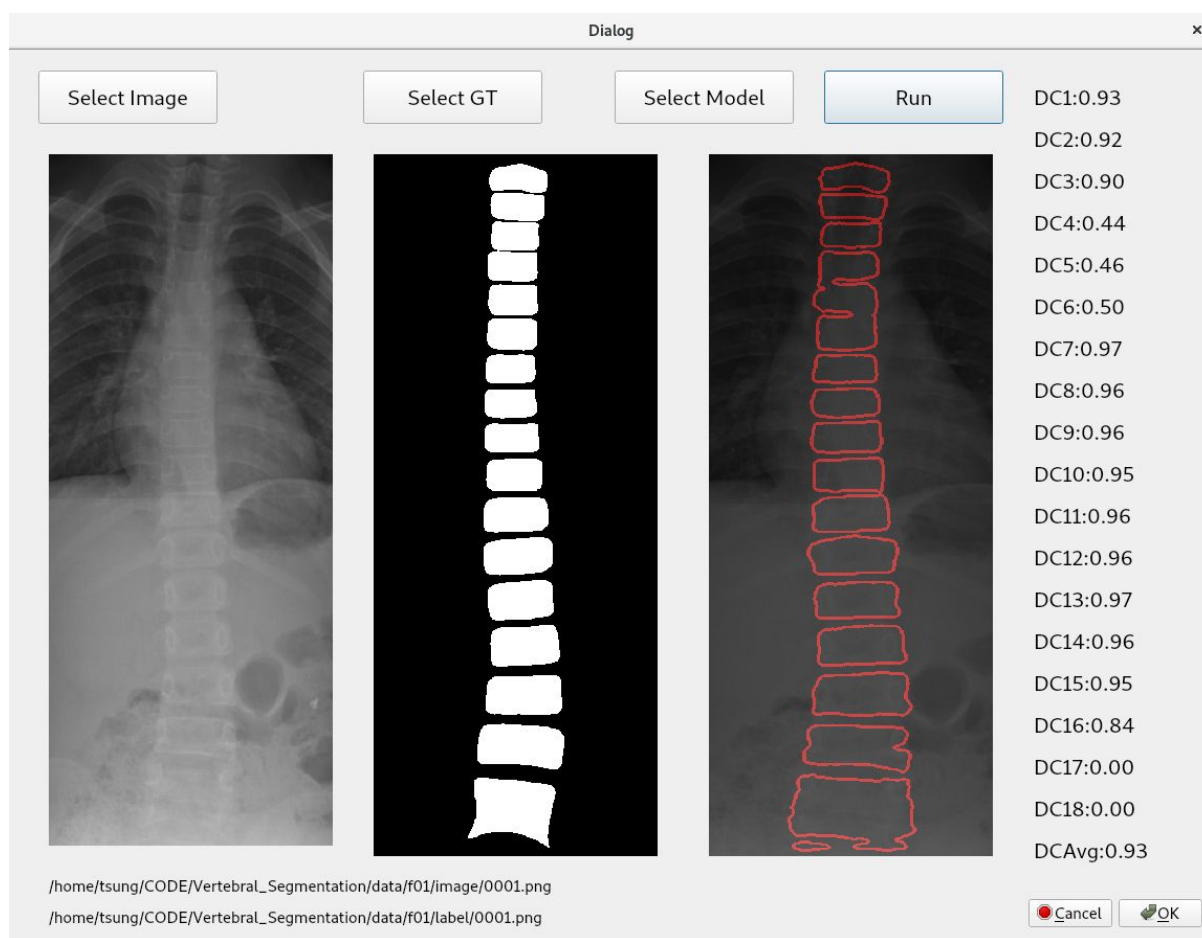
### 模型參數:

growth rate = 4 vs growth rate = 5



growth rate = 4 參數有1,812,337, growth rate = 5 參數有7,222,497, 目前更多的模型參數對於模型有更好的結果。

## 最後呈現



## D. Discussion

雖然模型的總 dice 很高，但在細節上並不理想，最後輸出很容易會將部份脊椎骨連在一起，表示目前model 也只是抓個大概而已，並沒有 robust。

我發現其他同學用普通的unet就可以有不錯的表現，我建立的 dense-unet 在這個任務上看來沒有太大的優勢，如果可以的話還是傾向用簡單的model就好。

## E. Future Work:

在這次的作業中並沒有對資料做太多的前處理，我認為是結果不理想的重大原因，上課的內容其實有很多方法可以用在資料上做強化的。即使不用那些演算法。