

1. Reproducibility of the results (6 pts)

我測試了許多次，因為我有做 randomized train dataset，所以每次結果雖然不太一樣，但是基本上都得到差不多的結果

2. Number of parameters: Please write the parameter count of your final selected model to the Kaggle competition (4 pts)

```
model = UNet()
total_params = sum(p.numel() for p in model.parameters() if p.requires_grad)
print(f"Total number of parameters: {total_params}")
del model
```

```
[08:02:32] [08:02:32] Total number of parameters: 27910146
```

27910146

3. The difficulty during training (8 pts)

跑的時間比較長，把資料放到 kaggle 上跑比較快

4. Briefly explain the structures of the models you are using: You are required to do analyses using FCN-8s and "U-Net (12 pts)

FCN-8s 基本結構：

卷積層與池化層：

類似於 VGG16 的結構，共有 5 個卷積區塊（block1 到 block5）。每個區塊包含多層卷積層，後接一個最大池化層，用於特徵提取與下採樣。

逐像素預測層：

在 block5 後加入 1x1 卷積，用於將高維特徵壓縮為分類的通道數。

跳躍連接：

從 block5、block4 和 block3 提取的特徵映射，經上採樣後進行逐像素相加，保留高層語義特徵與低層細節。

總上採樣：

將融合的特徵進行 8 倍的反卷積，使輸出尺寸恢復至輸入圖像大小。

U-Net 基本結構：

編碼器（Encoder）：

每個編碼器區塊包含：

兩層卷積層：3x3 卷積 + 批量正則化（BatchNorm）+ LeakyReLU 激活函數，負責提取特徵。

最大池化層（MaxPooling）：以 2 倍下採樣壓縮特徵圖的空間維度。

選擇性 **Dropout**：在瓶頸層之前加入 **Dropout** 防止過擬合。

功能：提取特徵並逐步降低圖像分辨率。

瓶頸層 (**Bottleneck**)：

設計類似於編碼器，但擁有更多通道 (1024 通道) 以捕捉更高層語義信息。

功能：作為整個網絡的信息瓶頸，保留最關鍵的特徵。

解碼器 (**Decoder**)：

每個解碼器區塊包含：

反卷積層 (**Transposed Convolution**)：以 2 倍上採樣恢復特徵圖的空間分辨率。

兩層卷積層：與編碼器類似，進一步細化恢復的特徵。

解碼器區塊的輸入：瓶頸或前一層的輸出，加上來自對應編碼器層的特徵 (跳躍連接)。

輸出層

使用 **1x1** 卷積層壓縮通道數至分類數量 (**num_classes**)。

將輸出上採樣至與輸入相同的大小，方便進行像素級分類。