Severstal Steel Defect Detection

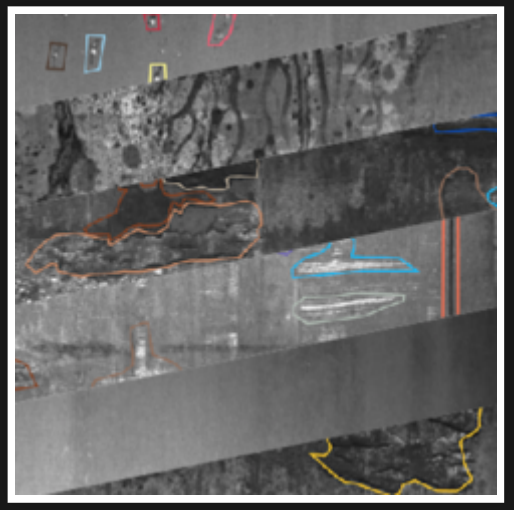
Team 12

B07502089 & B07902143

Introduction & Motivation

Story

- High frequency camera capturing images of steel surface, in order to detect 4 types of defects on steel



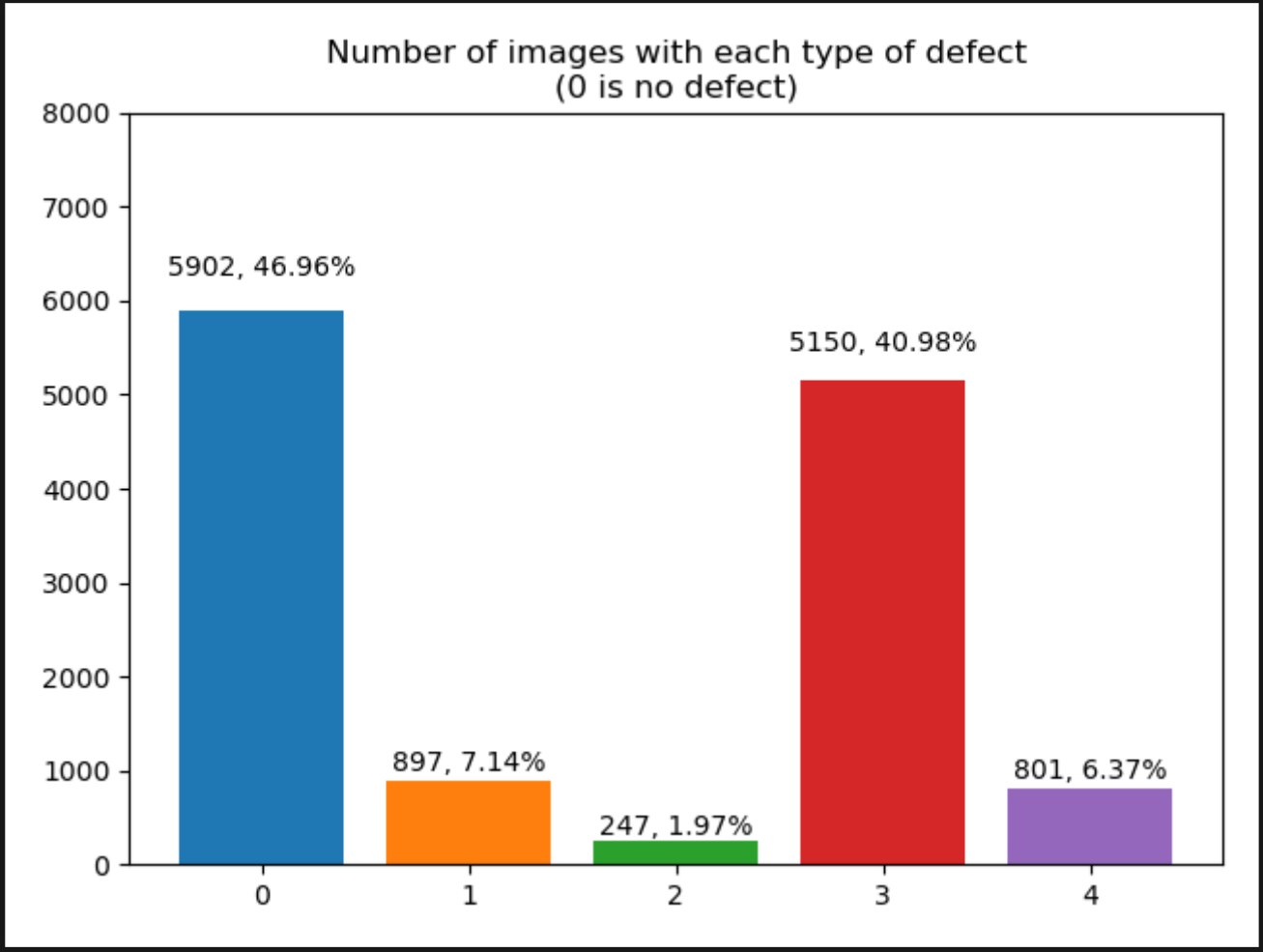
Goal and Data

- Goal: Multiclass Segmentation
- Training Data: 12568 images with dimension 1600x256x3
- Label: Each pixel marked class 1-4
- Loss function: Dice coefficient $\frac{2|X \cap Y|}{|X| + |Y|}$
    - X is predicted pixel
    - Y is groundtruith
    - Maximize to 1

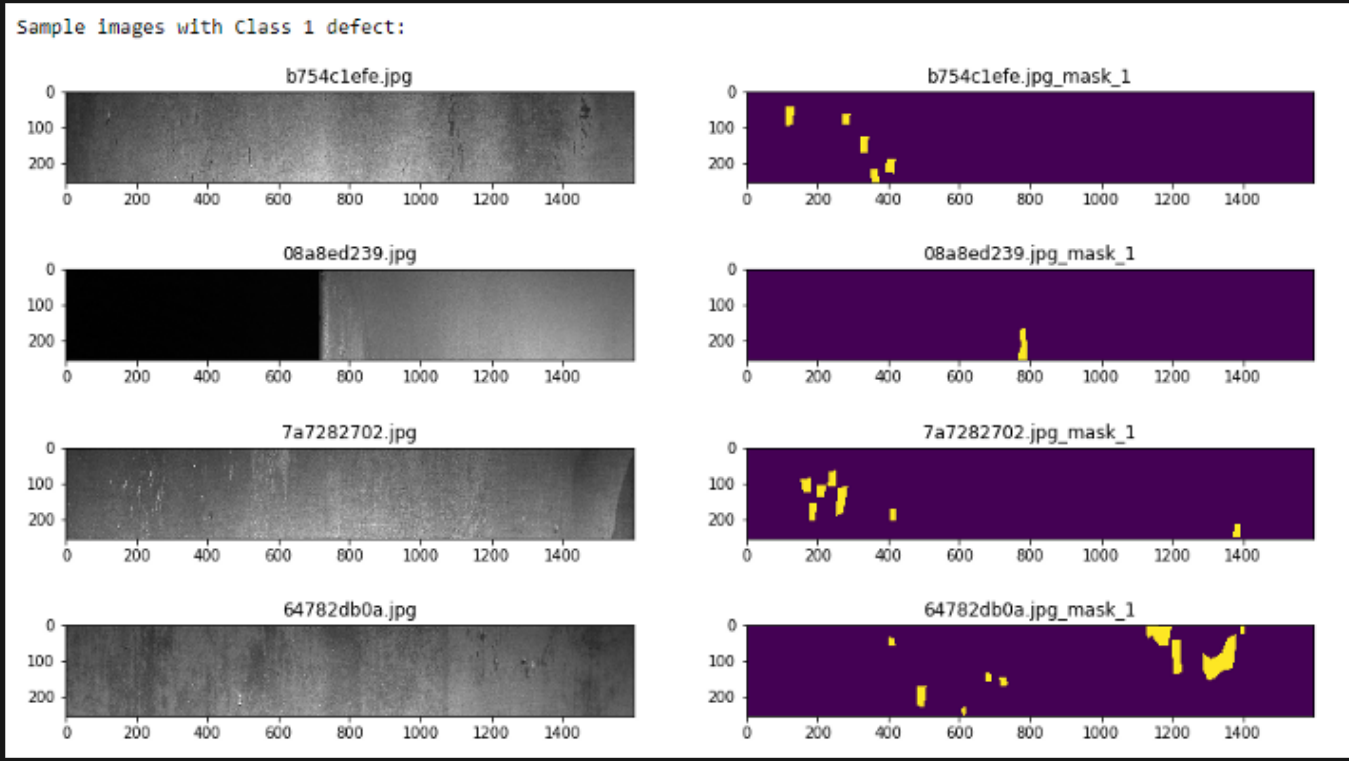
Data Analysis

- small data size
  - stratify to equally sample: 讓train set & val set 資料分布比例相同。
- 含有一種defect的image佔多數，有2種defect的image極少，有3種的&4種的: 0個

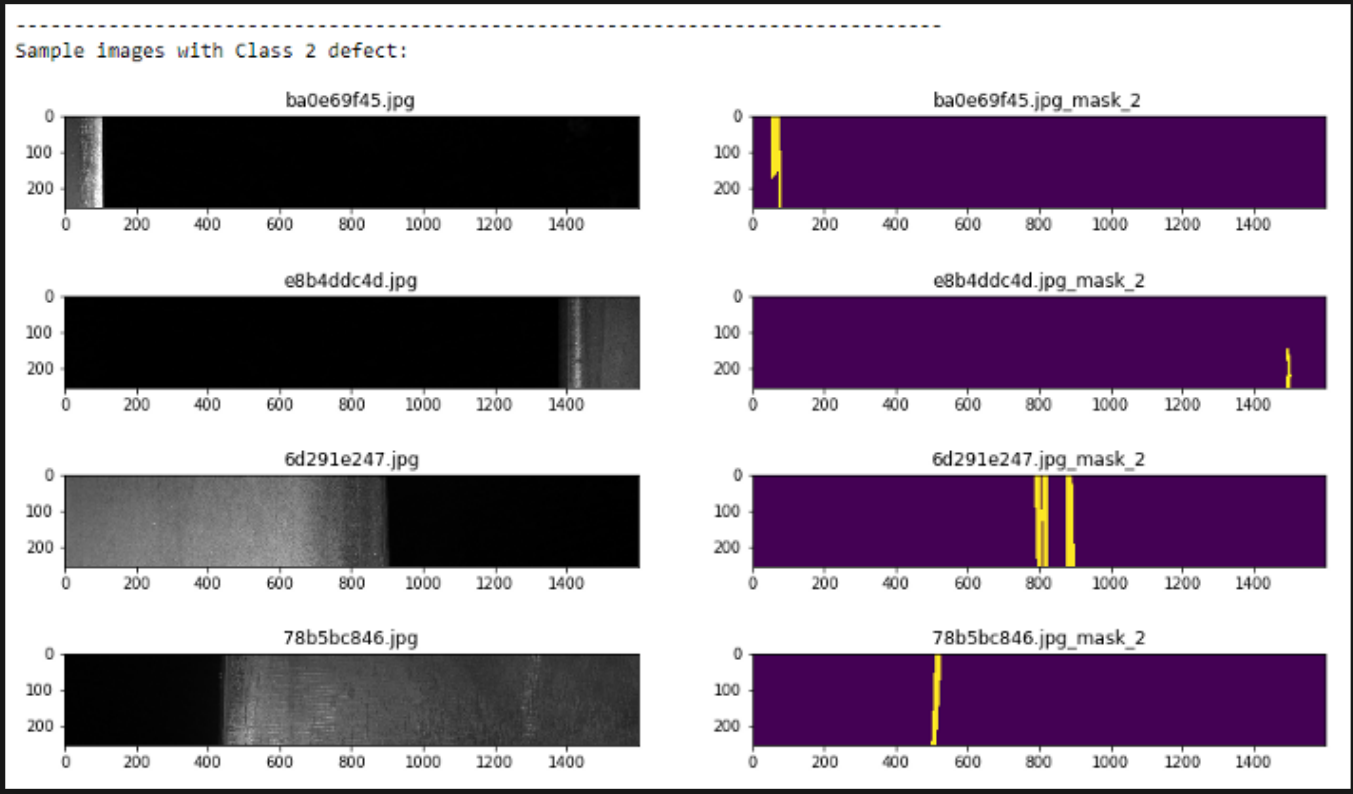
Data Analysis



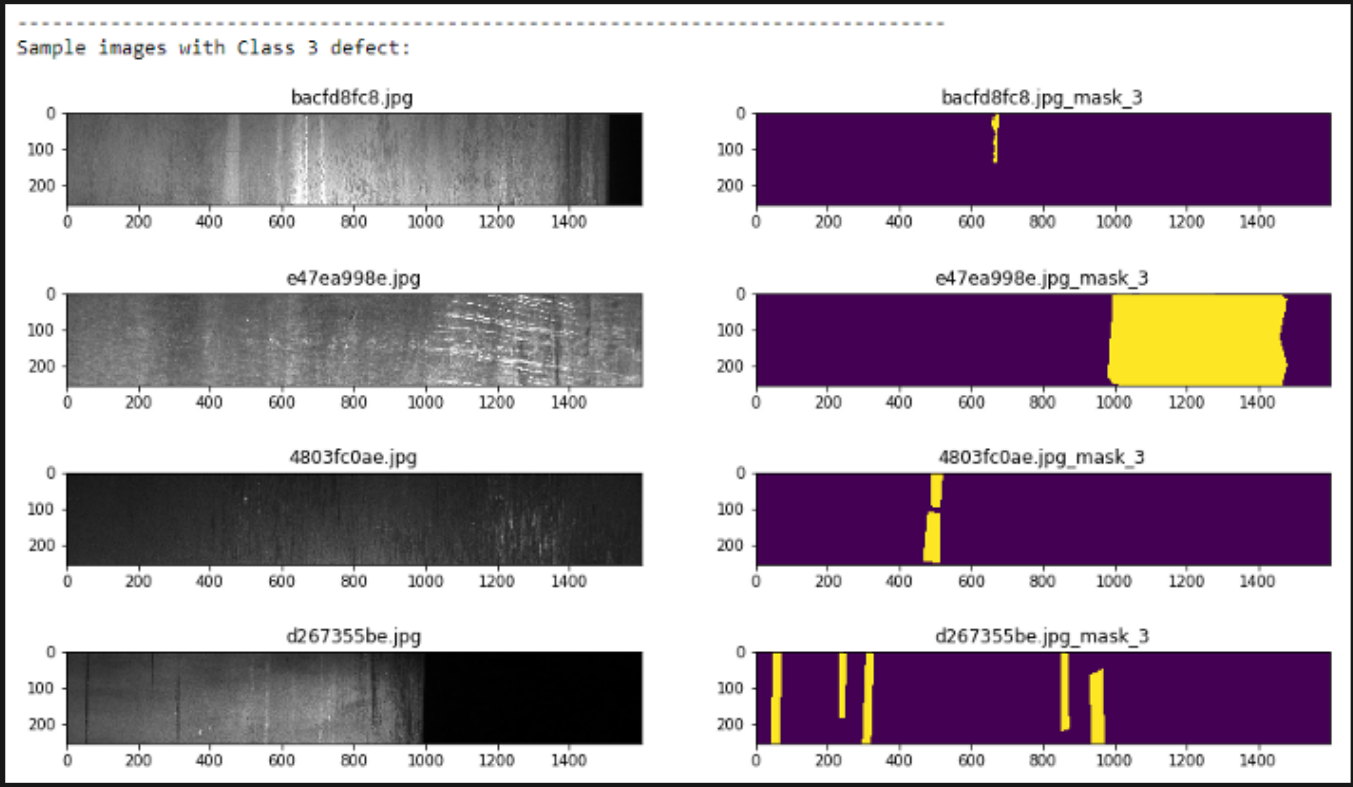
Different Defect Type



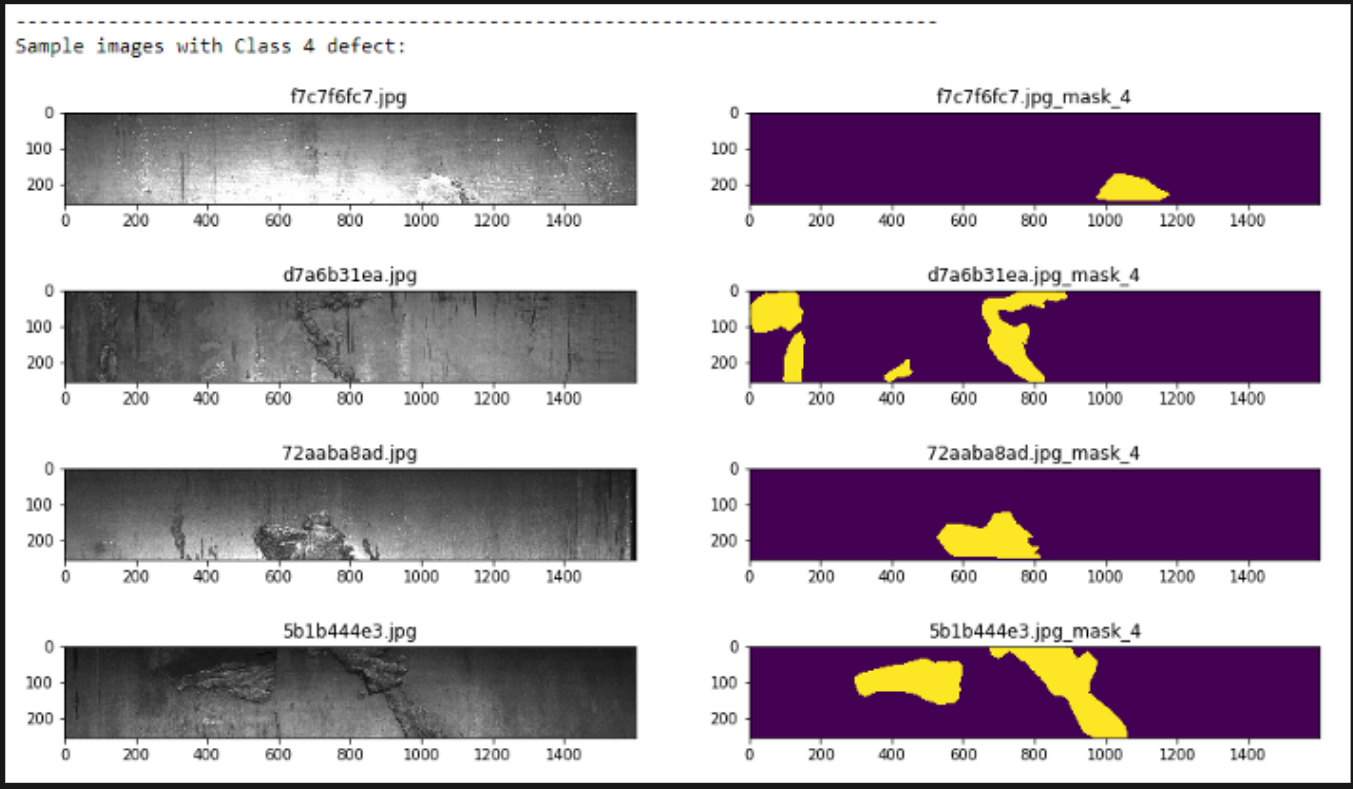
Different Defect Type



Different Defect Type



Different Defect Type



Motivation - How to Classify

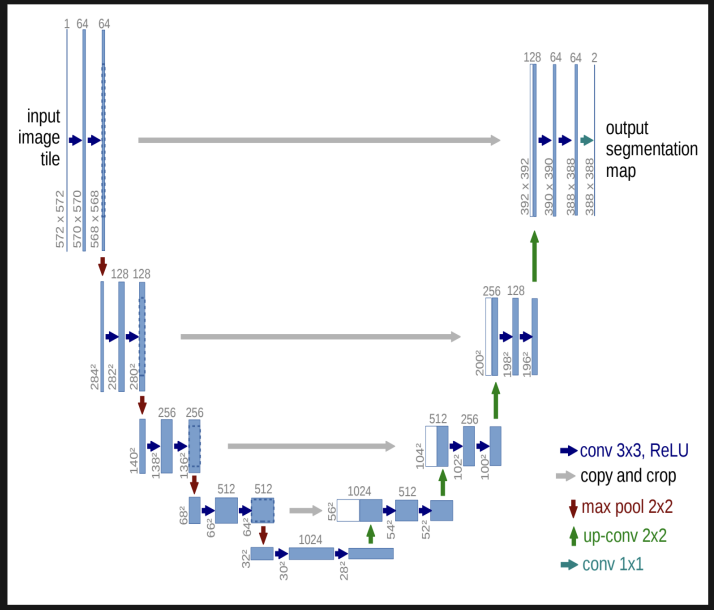
- Do binary classification first
  - Near half images without defect
  - By dice loss, false positive is expensive
- Then do 4-class classification
  - Determine type of defect

Motivation - How to Segment

- Perform multiclass **classification** on **each pixel**
  - CNN is capable
- Sementation Framework:
  - Unet
  - EfficientNetB1

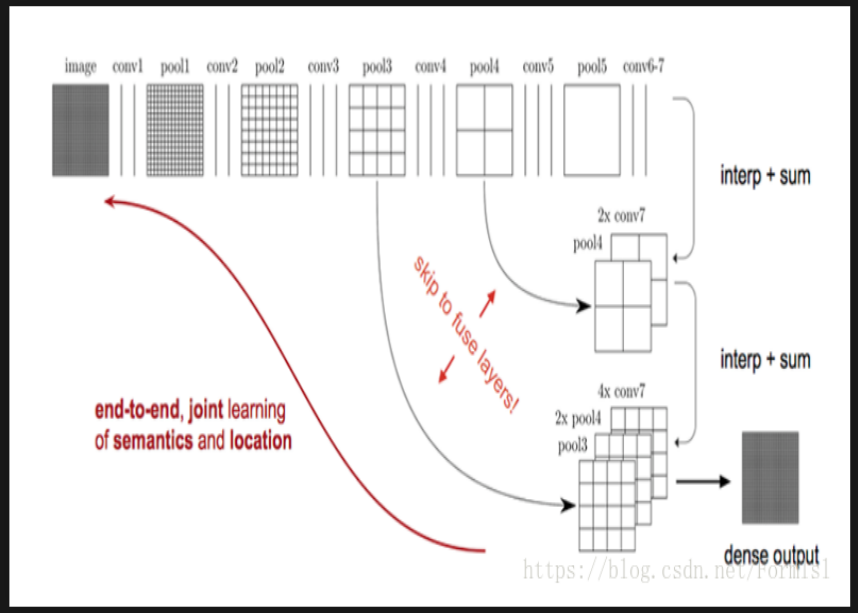
UNet

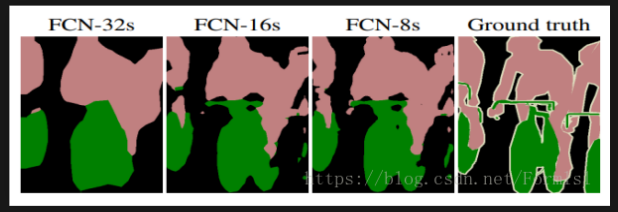
- No fully-connected layer
- Like CNN autoencoder: U-shaped



Unet

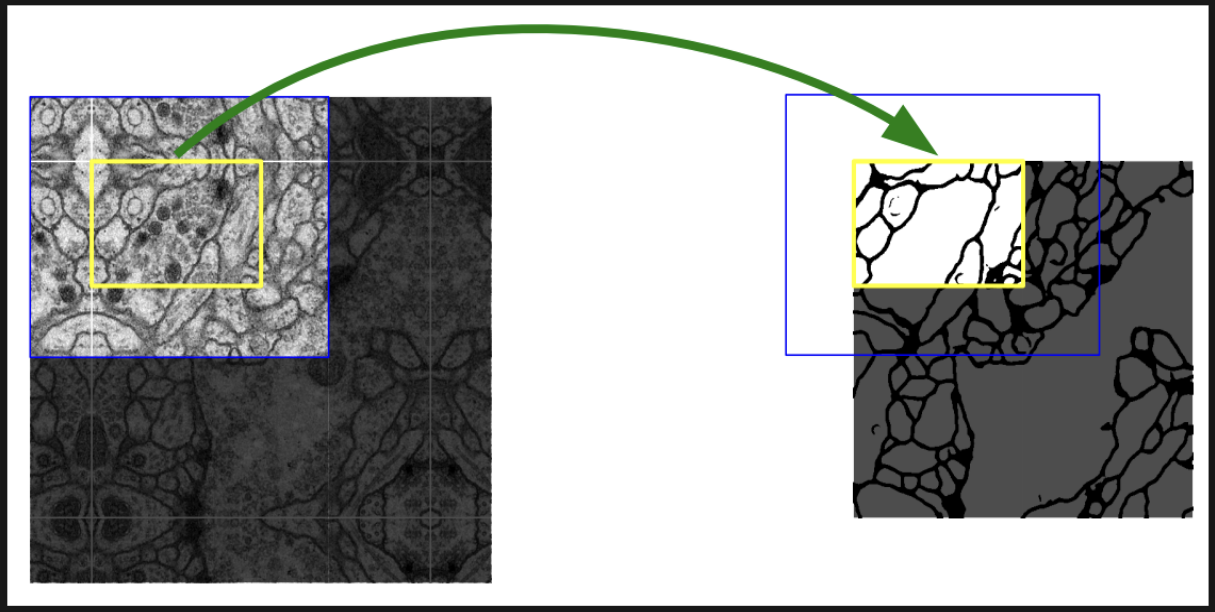
- Deconvolution: Combine previous pool





Unet

- Patching: Split an image into pieces won't affect result
  - As long as exists some overlap



Data Preprocessing

Augmentation on Training Data

- Normalization
  - Currently /255 only
- Zoom: 0.05
- Width-shift: 0.2
- Height-shift: 0.2
- Horizontal/Vertical flip

Augmentation on Training Data

- Random crop: 256x480
- Resize: 256x512
- Random brightness contrast

Model

Binary/Multiclass Classification Model

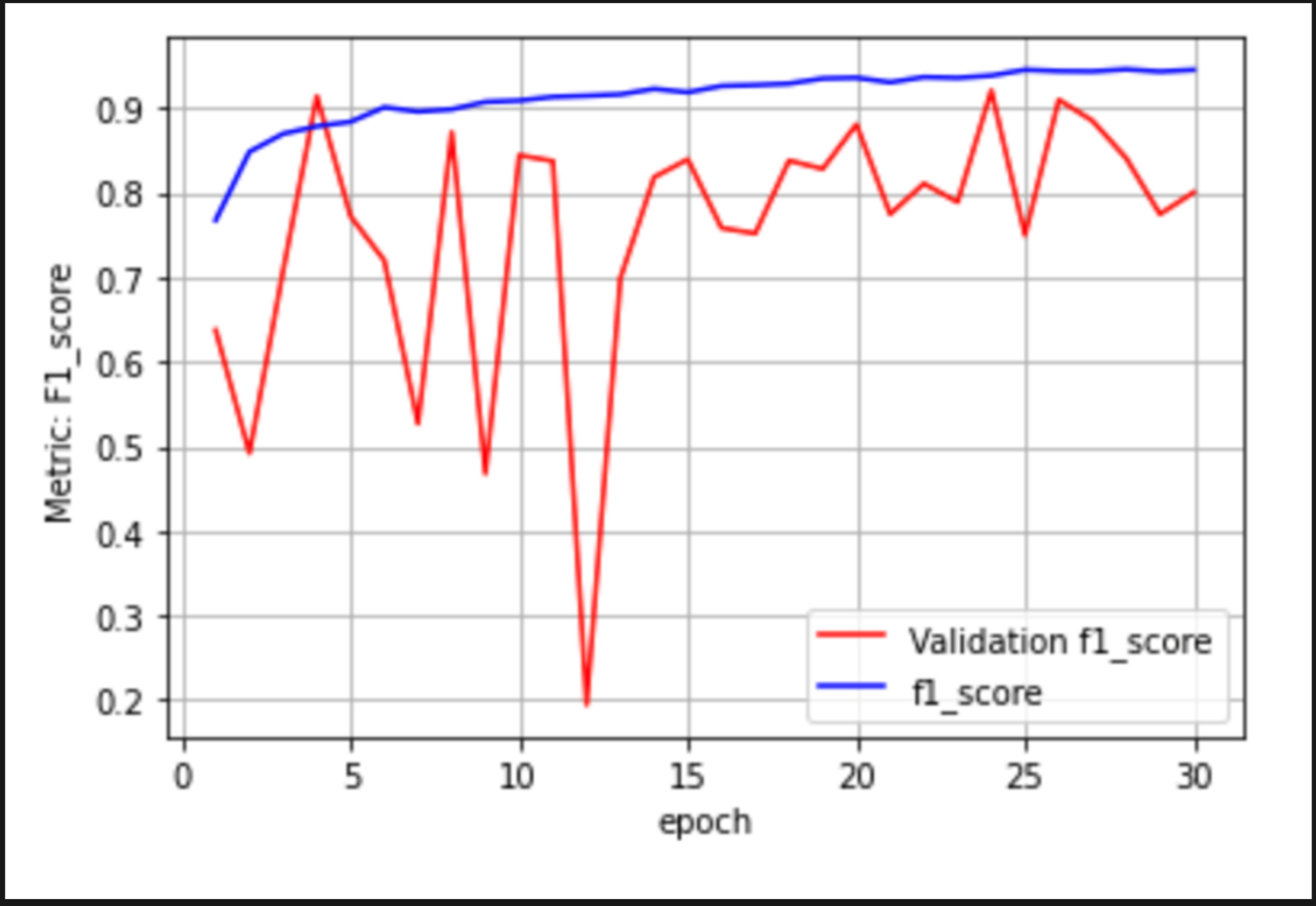
- 14-layer Xception CNN

Validation results:

binary_crossentropy 0.035537
acc 0.939563
f1_score_m 0.937449
precision_m 0.968126
recall_m 0.915408

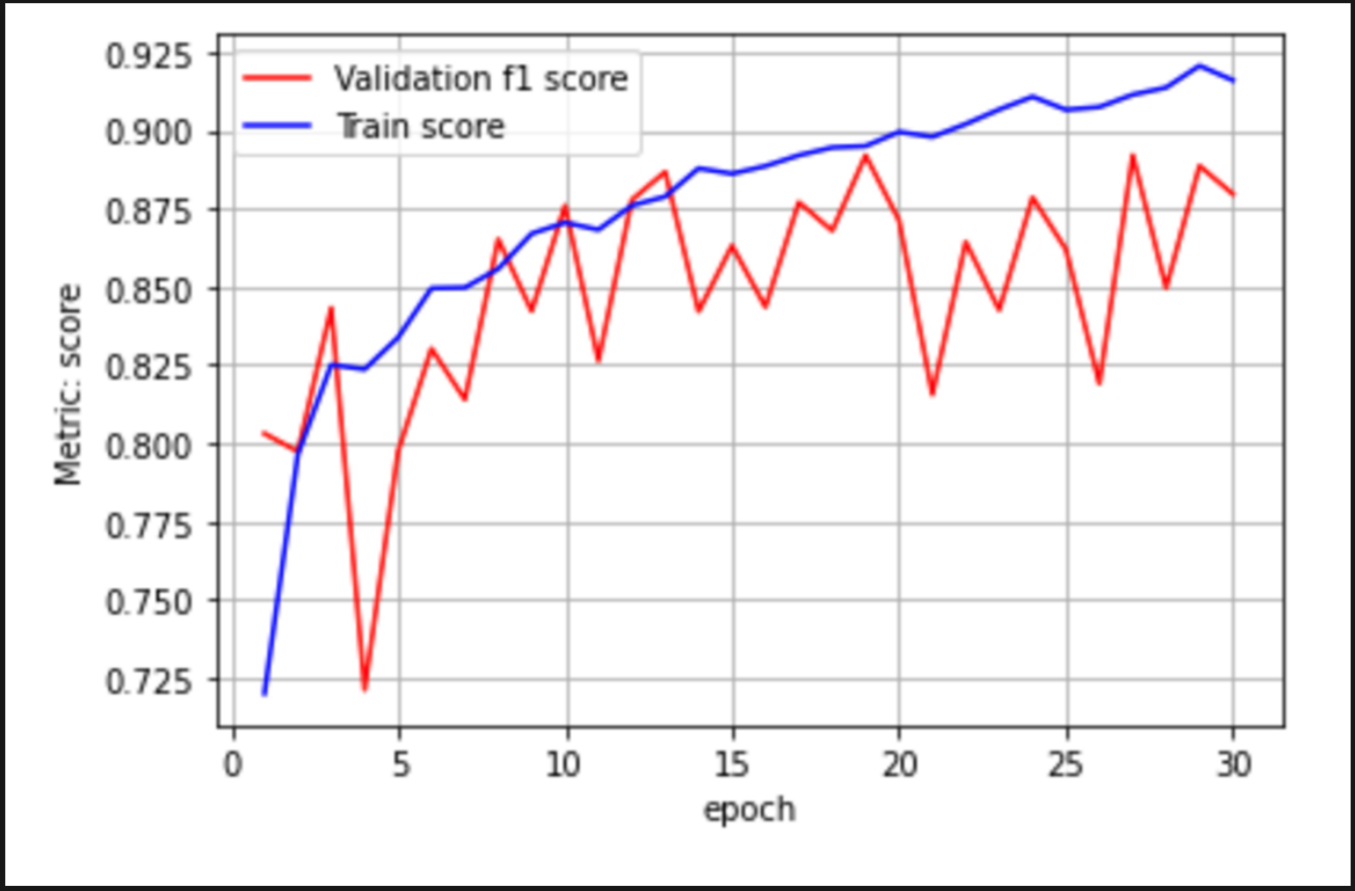
Binary Classification Model

Best val_f1_score: 0.92650



Multiclass Classification Model

Best val_f1_score: 0.90045

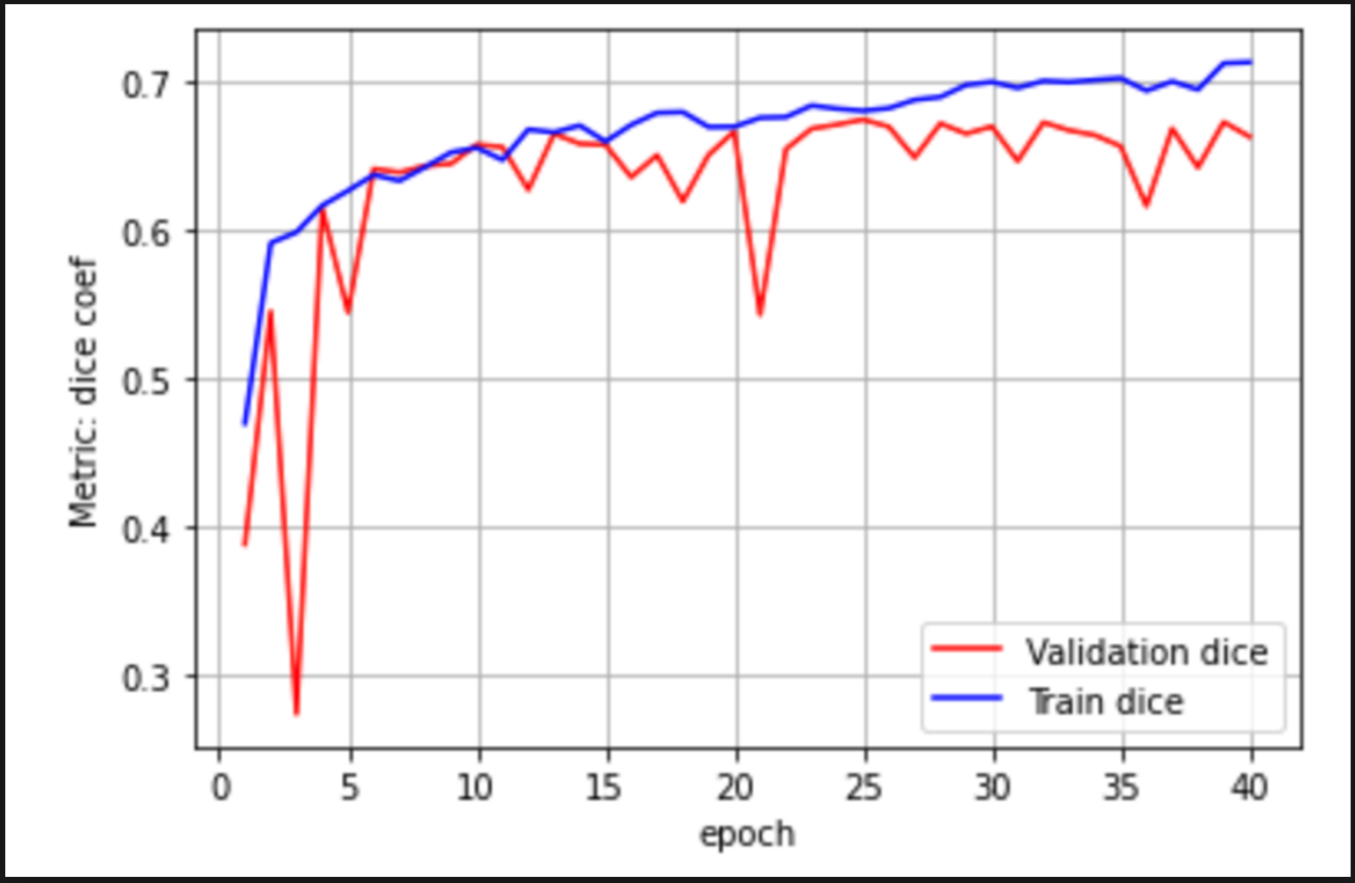


Segmentation Model

- One model for each type
- Pretrained EfficientNetB2

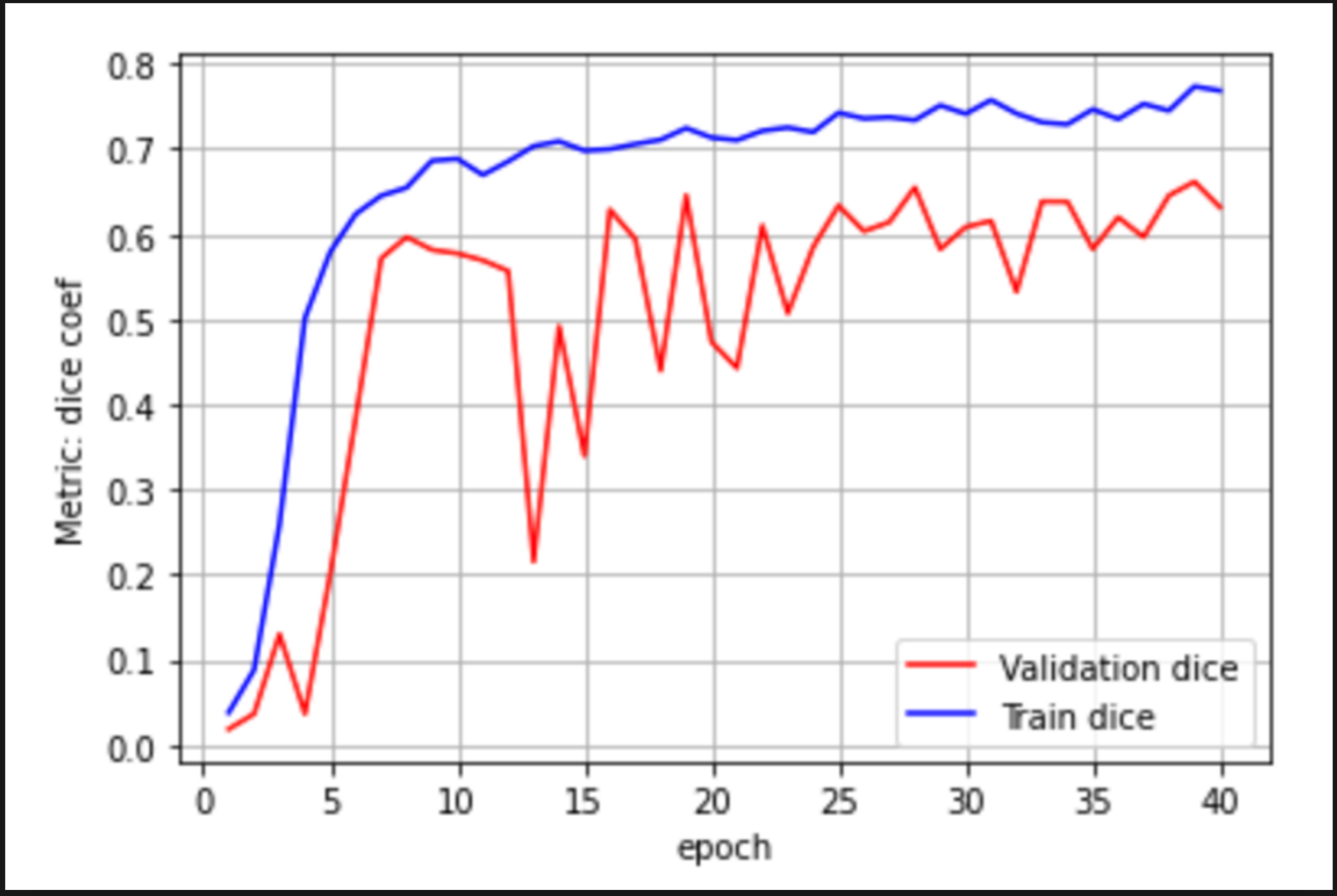
Segmentation Model - Type 1

Best dice coefficient: 0.67445



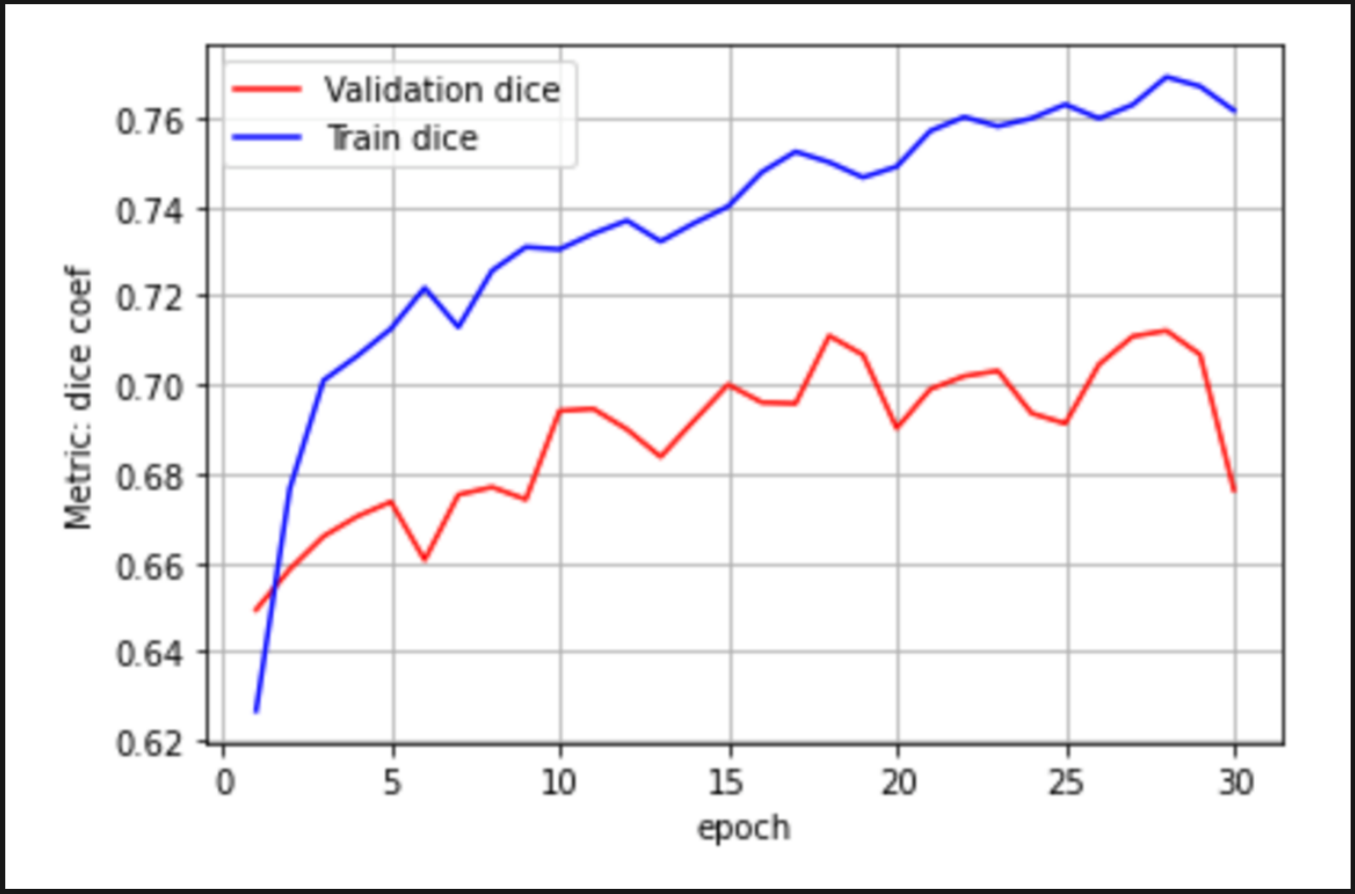
Segmentation Model - Type 2

Best dice coefficient: 0.66123



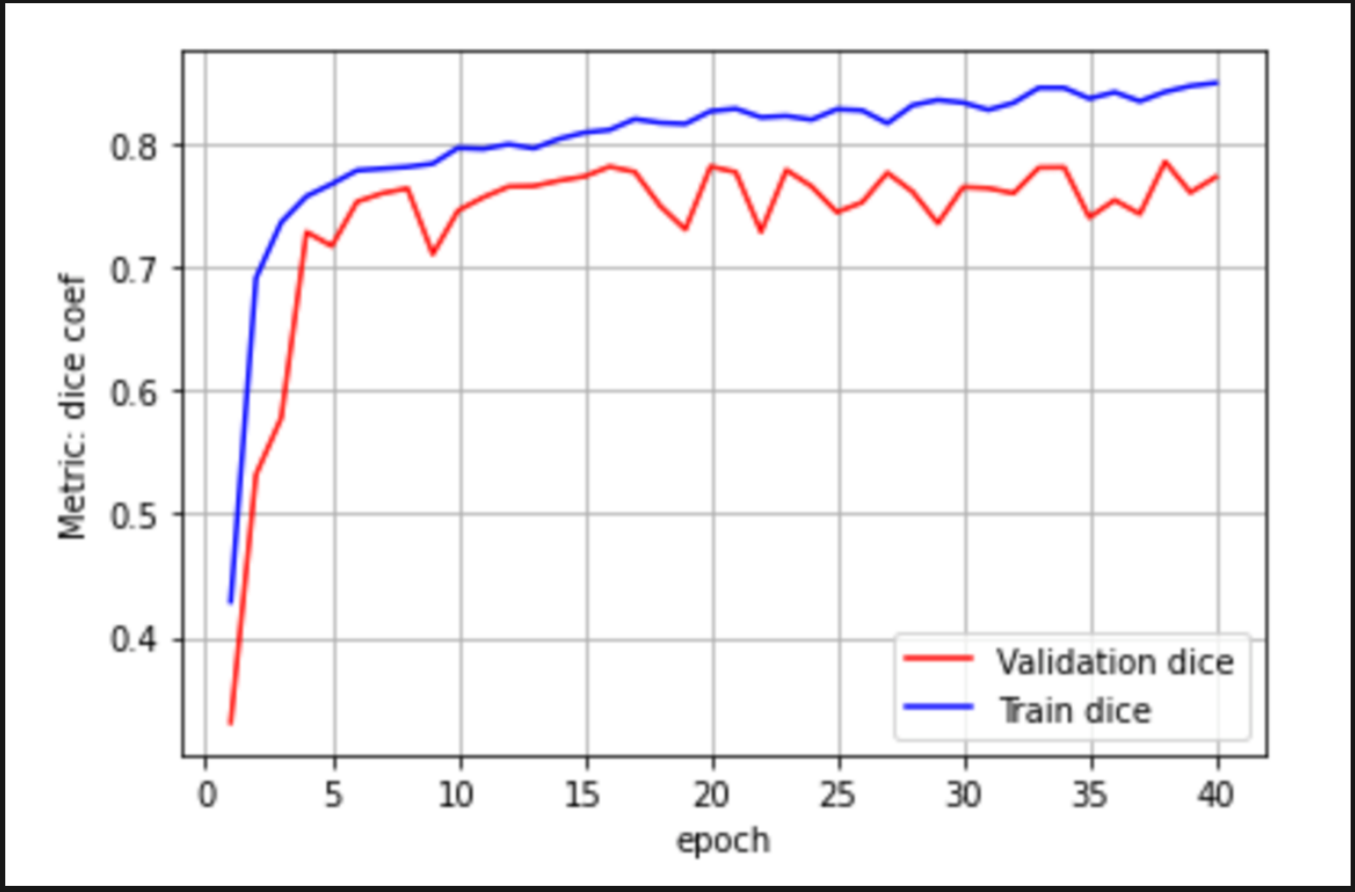
Segmentation Model - Type 3

Best dice coefficient: 0.71216



Segmentation Model - Type 4

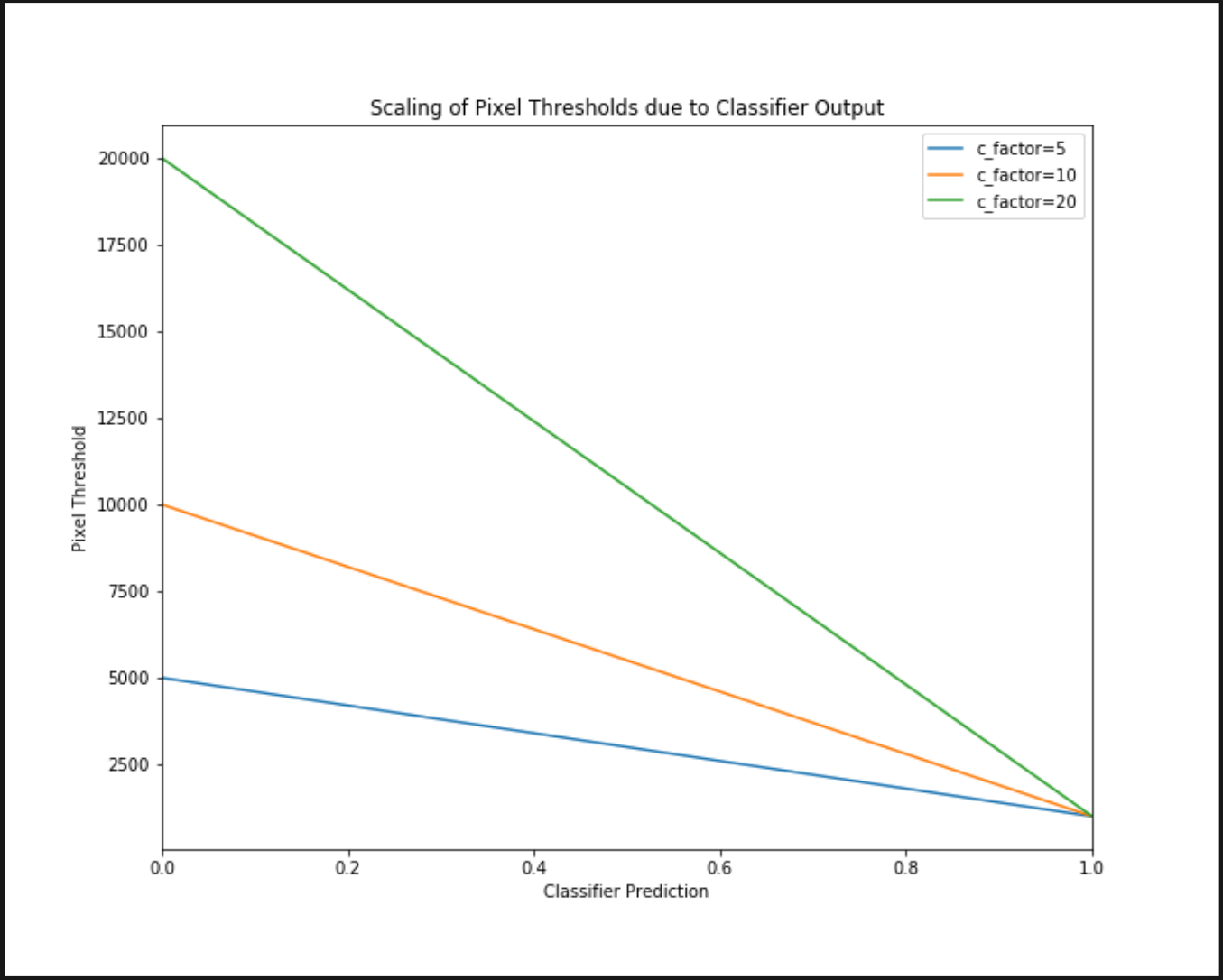
Best dice coefficient: 0.78626



Postprocessing and Inference

- Component domination
- Prediction threshold
  - Binary: [0.9, 0.85, 0.85, 0.85]
  - Multiclass: [0.5, 0.5, 0.75, 0.5]
  - 一張圖片有可能2種defect試過[0.4, 0.4, 0.4, 0.4]，但沒有比較好
- Detected #pixel threshold
  - [500,700,1100,2800]
  - Kaggle上其他人差不多都用[600,600,1000,2000]
  - Attempt: Determined by output of binary classifier

Postprocessing and Inference



Model Accuracy

Binary: 0.92650

Multi: 0.90045

segment_model_defect1 : 0.68155

segment_model_defect2 : 0.65966

segment_model_defect3 : 0.70743

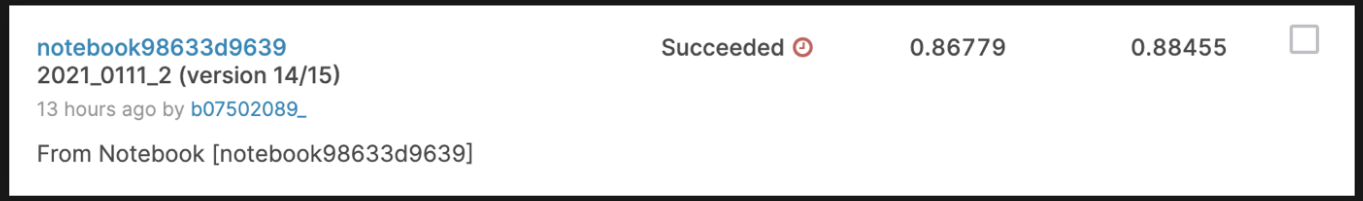
segment_model_defect4 : 0.77201

Performance

Kaggle best score (highest public)

Public: 0.88455

Private: 0.86779



Further Improvement

但沒時間做ㄌ...

- Pseudo Labeling
  - Choose the most confident prediction as pseudo label, add to training data
- Ensemble
- Try a single model rather than multilevel
- Augmentation
- Normalization

Reference

- https://medium.com/@guildbilla/steel-defect-detection-image-segmentation-using-keras-dae8b4f986f0
- https://github.com/khornlund/severstal-steel-defect-detection
- https://www.kaggle.com/c/severstal-steel-defect-detection/discussion/114254
- https://www.kaggle.com/khlevnov/imagedatagenerator-and-albumentations-without-pain
- https://github.com/rook0falcon/steel-defect-detection