Macau University Of Science and Technology

# Screening for novel 2019 coronavirus pneumonia using a deep learning system

Bachelor Thesis

*Author:*
Tianhao Xu,
1709853J-I011-0223

*Supervisor:*
Dr. Subrota Kumar Mondal

*A thesis submitted in fulfilment of the requirements for the degree of Bachelor of Science*

*in the*

Faculty of Information Technology

March 2021

MACAU UNIVERSITY OF SCIENCE AND TECHNOLOGY

# *Abstract*

Faculty of Information Technology

Bachelor of Science

**Deep Learning**

by Haoxi ZHANG, 1509853G-I011-0342

As of March 16, a total of 81,078 new cases of coronary pneumonia have been diagnosed nationwide and a total of 85,133 cases have been diagnosed abroad. The number of confirmed cases abroad has surpassed that at home. the global epidemic of COVID-19 is an indisputable fact.

In this battlefield without gunpowder, the killing power of stealth warfare is far more powerful than naked combat. For example, the new coronary pneumonia COVID-19 is so well camouflaged that it is difficult for doctors on the frontline battlefield to "see" it in the first place. The differences between it and influenza are precise.

First, both influenza and COVID-19 are contagious and both can cause respiratory illness. Typical flu symptoms include fever, cough, sore throat, aching limbs, headache, runny nose, nasal congestion, fatigue, and vomiting and diarrhea. The most common symptoms of new-onset coronary pneumonia are fever, cough, shortness of breath, and in 5

In other words, respiratory viruses cause similar symptoms, so it is difficult to distinguish COVID-19 from influenza from the symptoms themselves.

This study aims to build an early screening model to distinguish COVID-19 pneumonia from influenza A viral pneumonia and CT images of the lungs of healthy cases using deep learning techniques to help physicians make rapid diagnoses and improve diagnostic efficiency from an artificial intelligence perspective.

**Keywords**: Key words: coronavirus disease 2019 pneumonia, COVID-19, deep learning, computed tomography, convolution neural network, location-attention network.

# *Acknowledgements*

Acknowledgement here.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

graphicx

## 1.1 Introduction

COVID-19 causes severe respiratory symptoms and is associated with a relatively high rate of ICU admission and mortality. Current clinical experience in treating such patients shows that early detection of viral RNA in sputum or nasopharyngeal swabs by RT-PCR has a low positive rate. the CT imaging presentation of COVID-19 cases has its own characteristics and is different from that of other viral pneumonias such as influenza A viral pneumonia. Therefore, clinicians call for replacing nucleic acid testing with CT of the lungs as soon as possible as one of the early diagnostic criteria for this new type of pneumonia. With the rapid development of computer technology, digital image processing techniques have been widely used in the medical field, including organ segmentation, image enhancement and restoration, to support subsequent medical diagnosis. Deep learning techniques, such as convolutional neural networks (CNN) with strong nonlinear modeling capability, have also been widely used in medical image processing. Related studies were performed to classify benign and malignant On the diagnosis of pulmonary nodules.

### 1.1.1 Medical images in the medical field using image recognition technology

Image recognition is part of computer experience, the process of identifying and discovering objects or structures in a digital image or images. Computer experience is a broader term that encompasses methods of collecting, processing and analyzing information from real-life situations. This information is high-dimensional and creates mathematical or symbolic data in the form of conclusions. In addition to image recognition, computer vision includes event discovery, target recognition, education, image restoration, and television tracking.Also computer image recognition technology works by detecting salient regions that represent the part of the image or target that contains the most data. It does this by isolating and localizing the most informative parts or features in a selected image, while ignoring those parts of features that may not be of interest. More critically, recognition techniques of image features are also embarking on high-end applications in the field of medical development to help or assist doctors in medical diagnosis. In the current domestic research, AI-based computer-aided detection of chest radiography then begins to be useful. Chest radiography (chest X-ray or CXR) is an economical and easy-to-use medical imaging and diagnostic technique. This technique is the most commonly used diagnostic tool in medical practice and has an important role in the diagnosis of lung disease. Well-trained radiologists use automated recognition of chest X-ray images to detect diseases such as pneumonia, tuberculosis, interstitial lung disease and early lung cancer. Chest radiographs are widely used for the detection and diagnosis of lung diseases such as pulmonary nodules, tuberculosis and interstitial lung disease. However, correct interpretation of information is always a major challenge for physicians, and it is difficult for physicians to scientifically analyze the chest pictures of patients with their own knowledge alone, so scientific diagnosis is inevitably made with the help of deep learning image recognition. The overlap of tissue structures in chest X-rays greatly increases the complexity of interpretation.

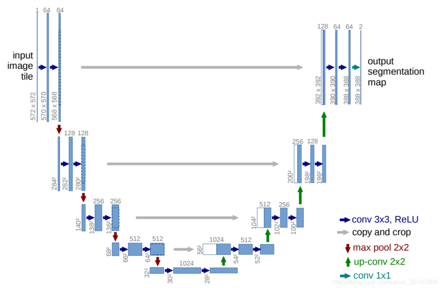
### 1.1.2 Approaches

In this section, i will introduce Semantic segmentation method and two neural network models Unet and Deeplab.

**Semantic segmentation method for pneumonia lesions** A method for segmenting a lesion of pneumonia, characterized in that it comprises. Predicting a lesion area on medical image data at a positive level based on an image semantic segmentation model; and counting the lesion area at each parallel level and calculating a lesion volume in combination with said lesion area at each parallel level. The image semantic segmentation model is built by the following training steps. Inputting all of the labeled or partially labeled sample data into the lesion segmentation model to obtain the predicted results of the output of said lesion segmentation model Based on the lesion detection frame predicted by said lesion detection model and the lung regions predicted by the lobe lung segmentation model, sifting out low-level false positive regions from the prediction results to obtain pseudo-labeling of said sample data and adding unlabeled sample data Reviewing said pseudolabel and labeling the labeled sample data to update said labeled sample data.

**Unet Neural Network Models** UNet was first published in MICCAI 2015, and in just 3 years, the number of citations has now reached 4070, which is enough to see its influence. And then it became the baseline for most of the tasks doing semantic segmentation of medical images, and also inspired a large number of researchers to think about U-shaped semantic segmentation networks. Nowadays, in natural image understanding, more and more semantic segmentation and object detection SOTA models have started to focus on and use U-shaped structures, such as Semantic Segmentation Discriminative Feature Network (DFN) (CVPR2018), Feature Pyramid Networks for Object Detection (FPN) for Object Detection (FPN) (CVPR 2017), etc.

UNet's structure, in my opinion, has two major features, the U-shaped structure and the skip-connection.




UNet's encoder is downsampled 4 times, and downsampled 16 times in total. Symmetrically, its decoder is also upsampled 4 times accordingly, and the advanced semantic features obtained by the encoder are restored to the resolution of the original image. Compared with FCN and Deeplab, UNet upsamples 4 times and uses skip connection at the same stage instead of directly supervising and loss backpropagation on the high-level semantic features, which ensures that the final recovered feature map incorporates more low-level features and also makes the different The 4 times upsampling also makes the segmentation map recover more fine information such as edges.

What are the characteristics of medical images?

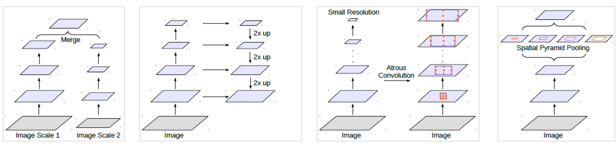
1. The semantics of the image is relatively simple and the structure is relatively fixed. If we do brain, we use brain CT and brain MRI, if we do chest X-ray, we use chest CT, if we do fundus, we use fundus OCT, all of them are imaging of a fixed organ, not the whole body. Since the organ itself is fixed in structure and not particularly rich

in semantic information, both high-level semantic information and low-level features become important (UNet's skip connection and U-shaped structure come in handy).

2. Small amount of data. It is relatively difficult to obtain data from medical images, and many competitions only provide less than 100 cases of data. Therefore, the model we designed should not be too big, and too many parameters can easily lead to overfitting. The number of parameters of the original UNet is about 28M (the number of UNet parameters for upsampling with transpose convolution is about 31M), and the model can be smaller if the number of channels is reduced exponentially. By reducing the number of channels twice, the number of UNet parameters is 7.75 M. By reducing it four times, the number of model parameters can be reduced to less than 2 M, which is very lightweight. Personally, I have tried using SOTA network for natural image semantic segmentation such as Deeplab v3 and DRN on my own project, and found that the effect is similar to UNet, but the number of parameters is much larger.

3. Interpretability is important. Since medical imaging is ultimately an aid to doctors' clinical diagnosis, it is not enough for the network to tell doctors whether a 3D CT is diseased or not, doctors have to further want to know where the lesion is, in which layer and at which location, is it segmented, can it find the volume, and at the same time, for the classification and segmentation results given by the network, doctors also want to know why, so some neural network interpretable It is useful to draw the activation map to see which areas of the network are activated.

**Deeplab Neural Network Models** DeepLabv3 further explores the null convolution, which is a powerful tool in semantic segmentation tasks: the filter field of view can be adjusted and the resolution of feature responses computed by convolutional neural networks can be controlled. To solve the problem of target segmentation at multiple scales, a cascade of null convolution or a parallel architecture of null convolution with different sampling rates is designed. In addition, the model comes with an ASPP (Atrous Spatial Pyramid Pooling) module, which can convolve features at multiple scales of acquisition to further improve the performance. It plays a very good effect for semantic segmentation of medical images




(a) Image Pyramid (b) Encoder-Decoder (c) Deeper w. Atrous Convolution (d) Spatial Pyramid Pooling

Figure 2. Alternative architectures to capture multi-scale context.

graphicx

# Chapter 2

# Experiment and Evaluation

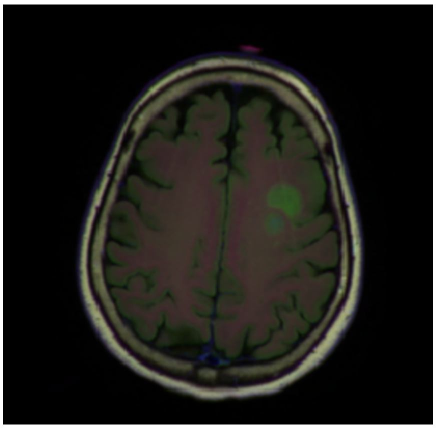
Firstly, I have completed the prediction of cancer which is based on the MRI dataset and Unet model. We can easily find that the unet model has a very good processing and prediction effect for CT diagram.

```
import os

import cv2
import matplotlib.pyplot as plt
import numpy as np
import torch

from pathlib import Path
from fastai.callbacks import *
from fastai.vision import *
```

```
# Specify the dataset path
path = Path('/kaggle/input/mri_dataset/MRI_Dataset/')
fnames = get_image_files(path / 'image')
lbl_names = get_image_files(path / 'label')
```

```
# Output the path of the current image
img_f = fnames[1020]
print(img_f)

# Show current image
img = open_image(img_f)
img.show(figsize=(5, 5))
```

```
/kaggle/input/mri_dataset/MRI_Dataset/image/TCGA_DU_A5TT_19980318_41.tif
```



```
# Get the path of the current image's label
get_y_fn = lambda x : path / 'label' / f'{x.stem}_mask{x.suffix}'
print(get_y_fn(img_f))
```

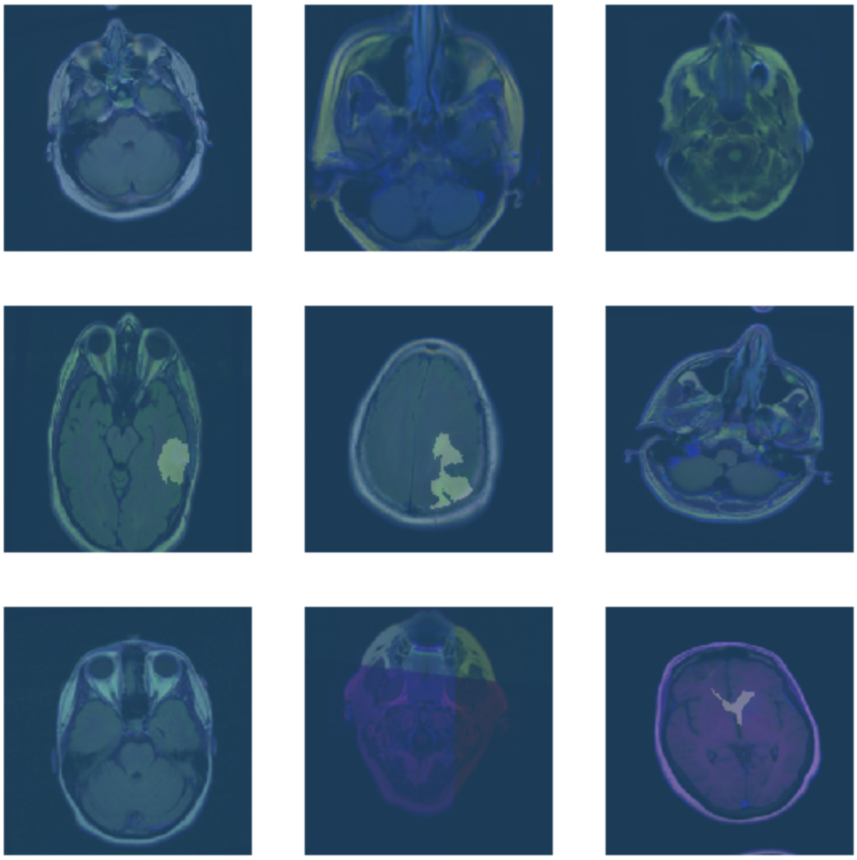
```
/kaggle/input/mri_dataset/MRI_Dataset/image/TCGA_DU_A5TT_19980318_41.tif
```

```
# Display the label of the current image
mask = open_mask(get_y_fn(img_f), div=True)
mask.show(figsize=(5, 5), alpha=1)
```

```
# Reading of images and their labels, pre-processing (normalization)
src_size = np.array(mask.shape[1:])
size = src_size # Image size (256, 256)
bs = 32      # batch size

class MySegmentationLabelList(SegmentationLabelList):
```

```
    def open(self, fn):
        return open_mask(fn, div=True)

class MySegmentationItemList(SegmentationItemList):
    _label_cls, _square_show_res = MySegmentationLabelList, False

data = (MySegmentationItemList
        .from_folder(path / 'image')
        .split_by_rand_pct(0.2)
        .label_from_func(get_y_fn, classes=['0', '1'])
        .transform(get_transforms(), tfm_y=True, size=size)
        .databunch(bs=bs, path='/kaggle')
        .normalize(imagenet_stats))
```

```
# Show partial data
data.show_batch(3, figsize=(8, 8), alpha=0.5)
```



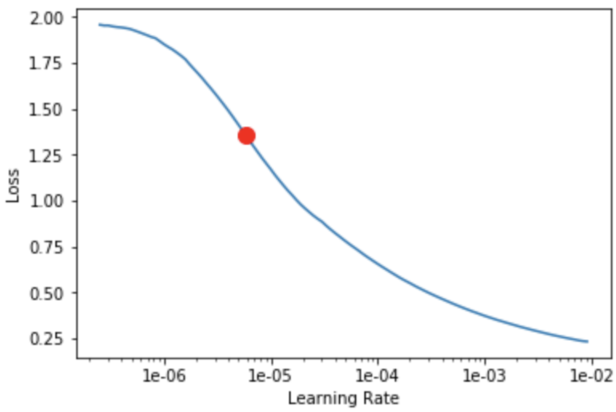
```
# Define the model as a unet model with resnet34 as the front end.
```

```
learn = unet_learner(data, models.resnet34, self_attention=True,
metrics=dice, wd=1e-2).to_fp16()
```

```
Downloading: "https://download.pytorch.org/models/resnet34-333f7ec4.pth"
to /root/.cache/torch/checkpoints/resnet34-333f7ec4.pth
100%|██████████| 87306240/87306240 [00:02<00:00, 35290918.60it/s]
```

```
# Find the best learning rate
learn.lr_find(num_it=200)
learn.recorder.plot(suggestion=True)
```

```
LR Finder is complete, type {learner_name}.recorder.plot() to see the gr
aph.
Min numerical gradient: 5.75E-06
Min loss divided by 10: 9.12E-04
```

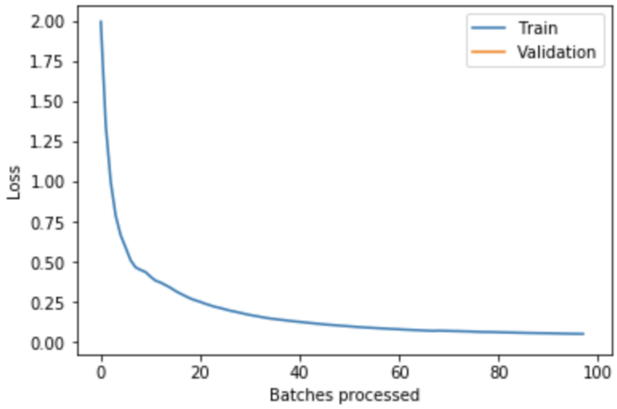



```
# Training model
learn.fit_one_cycle(1, max_lr=slice(6.92e-04, 1e-03), pct_start=0.7,
callbacks=[SaveModelCallback(learn, every='improvement', monitor='valid
```

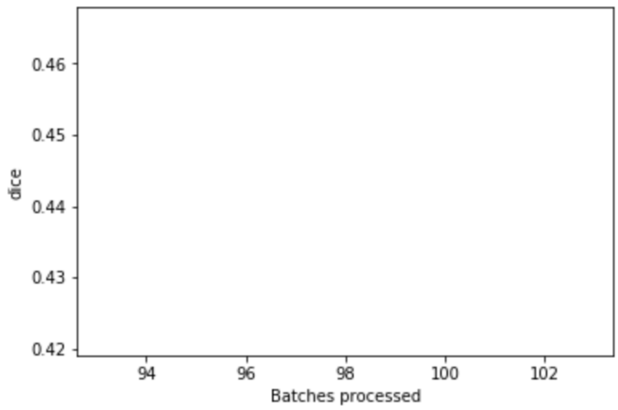
| epoch | train_loss | valid_loss | dice | time |
|---|---|---|---|---|
| 0 | 0.051976 | 0.024234 | 0.443472 | 01:40 |

```
Better model found at epoch 0 with valid_loss value: 0.02423426136374473
6.
```

```
# Plot loss curves
learn.recorder.plot_losses()
```



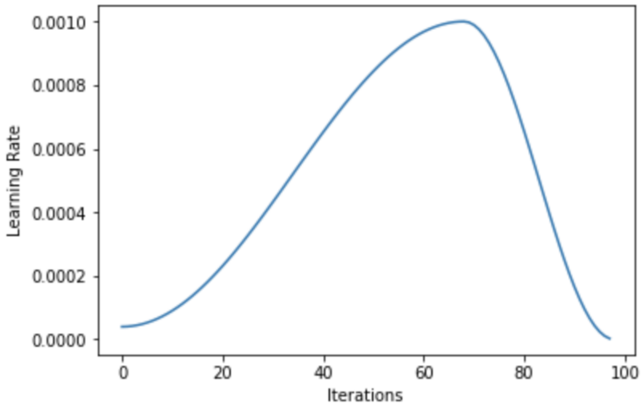

```
# Plot metrics curves (dice coefficients)
learn.recorder.plot_metrics()
```




```
# Plot the learning rate curve
learn.recorder.plot_lr()
```

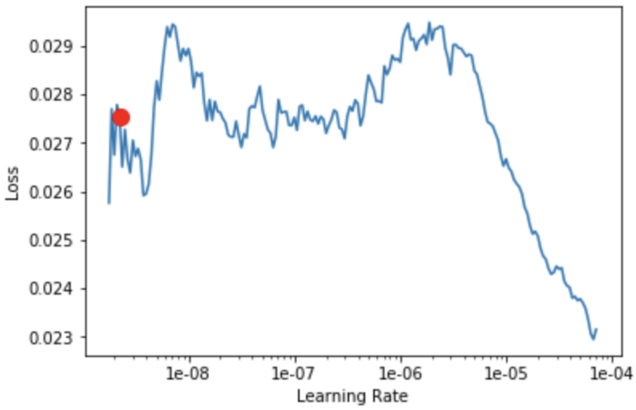
```
# Load the trained optimal model and unfreeze all layers. After that
the model will be fine-tuned
learn.load('model')
learn = learn.to_fp16()
learn.unfreeze()
```

```
# Find the optimal learning rate
learn.lr_find(start_lr=1e-9, end_lr=1e-4, num_it=200)
```




```
learn.record.plot(suggestion=True)
```

```
LR Finder is complete, type {learner_name}.recorder.plot() to see the gr
aph.
Min numerical gradient: 2.24E-09
Min loss divided by 10: 6.68E-06
```

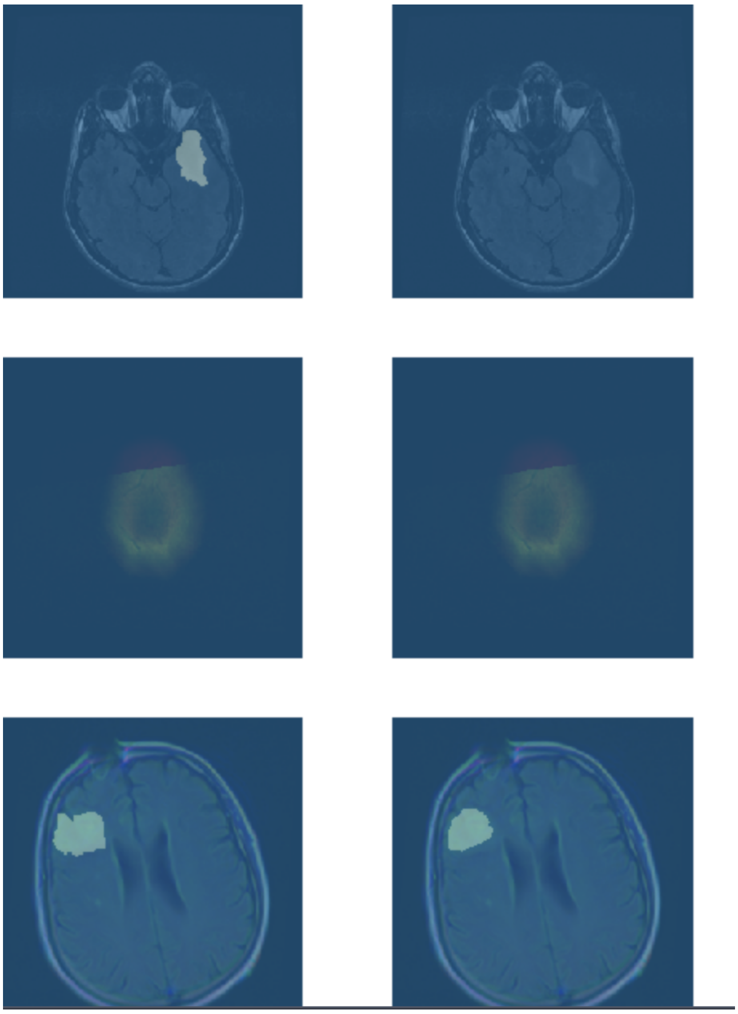


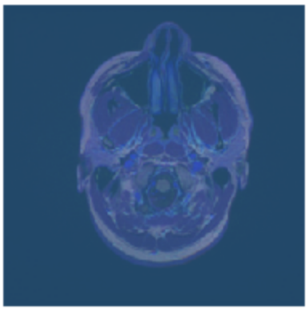

```
# Model fine-tuning using small learning rate
learn.fit_one_cycle(5, max_lr=slice(6.68e-06, 8e-09), pct_start=0.8,
callbacks=[SaveModelCallback(learn, every='improvement', monitor='valid
```

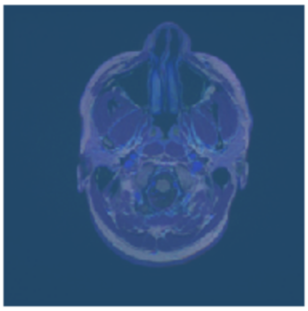
```
# Load the fine-tuned best model and show the prediction results
learn.load('model')
learn = learn.to_fp16()
learn.show_results(alpha=0.6)
```

| epoch | train_loss | valid_loss | dice | time |
|---|---|---|---|---|
| 0 | 0.028170 | 0.024310 | 0.430269 | 01:39 |
| 1 | 0.027263 | 0.024113 | 0.431849 | 01:39 |
| 2 | 0.027993 | 0.024086 | 0.435742 | 01:39 |
| 3 | 0.027391 | 0.023817 | 0.450601 | 01:39 |
| 4 | 0.027425 | 0.023960 | 0.444818 | 01:39 |

```
Better model found at epoch 0 with valid_loss value: 0.02430983819067478
2.
Better model found at epoch 1 with valid_loss value: 0.02411304228007793
4.
Better model found at epoch 2 with valid_loss value: 0.02408559247851371
8.
Better model found at epoch 3 with valid_loss value: 0.02381689473986625
7.
```







After completing the above experiments, I went to use deeplab V3 to retrain the dataset related to covid-19 semantic segmentation.

The following results were obtained

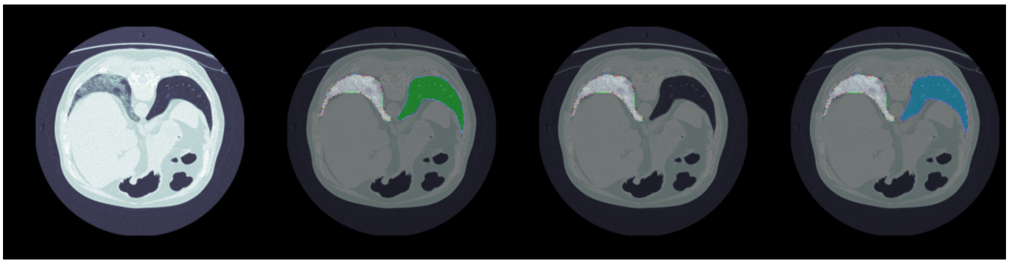


FIGURE 2.1: Label1

FIGURE 2.2: Label2

FIGURE 2.3: Randomly selected from the training data

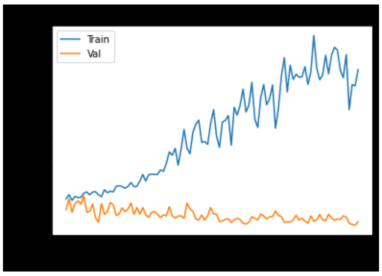



FIGURE 2.4: Curve of dice with epoch

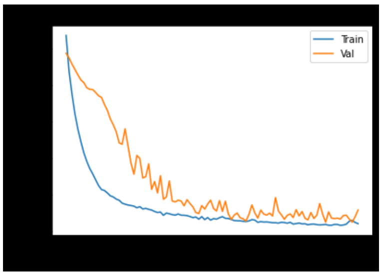



FIGURE 2.5: Loss curve

FIGURE 2.6: Prediction1

FIGURE 2.7: Prediction2

FIGURE 2.8: Prediction3

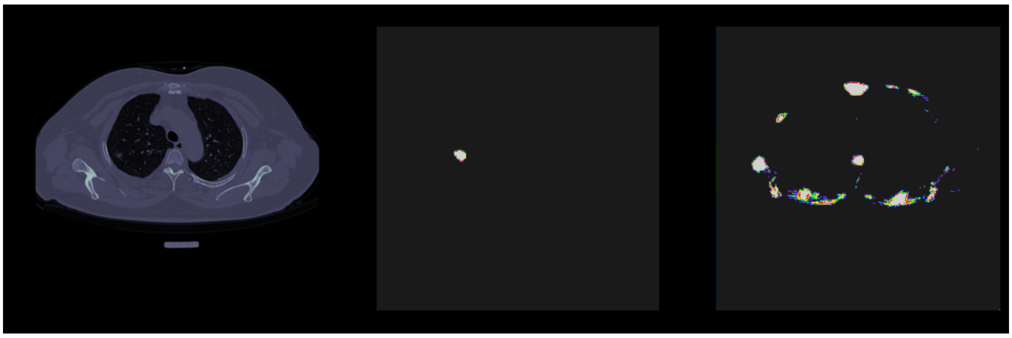


FIGURE 2.9: Prediction4

FIGURE 2.10: Prediction5

FIGURE 2.11: Prediction6

The effect of training with model deeplab was not good, and then I also used unet to train, but the effect was also not very good. Because there are fewer open source datasets for semantic segmentation about covid-19 The amount of data on covid-19 is just about 30 pictures of this dataset. So now I am trying to use Data enhancementhope to enhance the dataset to several hundred data so that improve the effect.

# Chapter 3

# Post-Program prospect

First I would go for data enhancement and see how well it works after augmentation, if that does not work, I would consider using a convolutional neural network for the covid-19 classification task. Since training on small datasets tends to cause overfitting, we can use a transfer learning based network training method. A dataset with sufficient data is used for model pre-training. To solve the problem of model degradation caused by poor data quality, a softmax loss function constrained network training method based on feature normalization is used. By adding a feature normalization layer and a scale amplification layer to the network, the problem of focusing on high quality training images while ignoring low quality training images in network training can be avoided.

# Bibliography

1.Abnormal respiratory patterns classifier may contribute to large-scale screening of people infected with COVID-19 in an accurate and unobtrusive manner.

2.Prediction of criticality in patients with severe Covid-19 infection using three clinical features: a machine learning-based prognostic model with clinical data in Wuhan

3.Deep Learning System to Screen Coronavirus Disease 2019 Pneumonia

**Resume**

*Name*: XYZ
*Gender*: Male
*Email*:
*Education*:

*Work Experience*: