COVID-19 Chest CT Image Segmentation of Anomalies Using Fully Convolutional DenseNet with Max Unpooling

Joe Liu [1] **Qifeng Tan** [1]

Abstract

Many deep learning techniques have been applied on semantic segmentation of COVID-19 CT frames to identify regions of anomalies, but FC-DenseNet has not been investigated. In this paper, FCDenseNet is implemented on this specific segmentation task, in comparison with a new method named FCDenseNet V2 (i.e., FCDenseNet with max unpooling). Under different segmentation scenarios, FCDenseNet V2 has shown a better performance or a better stability or both on different classes (i.e., targets) in different sizes over FCDenseNet.

1. Introduction

COVID-19(Wu et al., 2020), as one of the major catastrophes in the history of mankind, is a severe respiratory disease, causing thousands of deaths daily. Lack of early diagnosis and accurate assessment of COVID-19 easily results in ineffective therapy, which not only slows down the process of recovery for patients but also hinders the infection control of the disease. Reverse transcription-polymerase chain reaction (RT-PCR) has been widely used for diagnosing COVID-19 at an early stage. However, due to the shortage of rapid and highly sensitive RT-PCR tests(Hope et al., 2020), a screening or diagnostic approach via imaging, such as computed tomography (CT), has been considered as an alternative tool for the diagnosis. The common anomalies have been identified by radiologists in chest CT images, including ground-glass opacity (GGO), consolidation(Chung et al., 2020), and rare characteristics, such as pericardial effusion and pleural effusion(Caruso et al., 2020), with GGO being a common feature among all chest CT images. Developing a reliable tool for semantically segmenting chest CT images of COVID-19 patients to achieve human-like performance would help to identify and quantify those abnormalities automatically, which can further facilitate early diagnosis.

[1]University of Wisconsin-Madison. Correspondence to: Joe Liu <yliu945@wisc.edu>, Qifeng Tan <qtan34@wisc.edu>.

Fully convolutional DenseNet (FCDenseNet)(Jégou et al., 2017), as an extension of DenseNet(Huang et al., 2017), inherits the advantages of DenseNet, including feature reuse, parameter efficiency and implicit deep supervision. It also improves the state-of-the-art performance on urban scene understanding datasets without additional post-processing, pretraining, or temporal information. However, it has not been applied on semantic segmentation of chest CT images of COVID-19 patients. *In this work, FCDenseNet has been primarily investigated for segmenting GGO, consolidation, and other lung areas*. In FCDenseNet, some information from earlier dense blocks can be lost in the downsampling path when coarse semantic features are extracted, due to the pooling operation. To overcome this issue, skip connections can be incorporated in the network to pass this information in the downsampling path to the upsampling path. Similarly, max unpooling in SegNet(Badrinarayanan et al., 2017) reuses pooling indices in the downsampling path, which can not only capture and store boundary information but also reduce the loss of spatial resolution of the feature maps. Due to the vital role that spatial information of pixels plays in image segmentation tasks, *max unpooling has been proposed in our work to replace transposed convolution as the upsampling technique to further enforce connectivity between downsampling and upsampling path on top of skip connections*. FCDenseNet has been also compared to our proposed *FCDenseNet V2* (i.e., FCDenseNet with max unpooling) on the task of segmenting anomalies in chest CT images to demonstrate its improvements.

2. Related works

Deep learning (DL) has been one of the most effective approaches to automatically identify regions of anomalies from tissues or organs and to achieve a human-like performance on various types of medical images, like CT scans and MRI images. For example, an innovative DL approach has been proposed for the automated diagnosis of lung cancer in CT images, with the sensitivity of 95.26%, specificity of 96.2%, and accuracy of 96.2%(Lakshmanaprabu et al., 2019). Moreover, a hybrid network of U-Net(Ronneberger et al., 2015) and SegNet has been applied to segment brain tumors from MRI images, achieving an accuracy of 0.99

(Daimary et al., 2020). The typical segmentation architecture comprises a downsampling path responsible for extracting coarse semantic features, followed by an upsampling path responsible for recovering the input image resolution. Both U-Net and SegNet have been applied on semantic segmentation of chest CT images of COVID-19 patients(Saood & Hatem, 2021), but FCDenseNet has not been implemented on such a task yet. A list of wildy-used DL frameworks in image segmentation will be discussed here to further introduce our inspiration for *FCDenseNet V2*.

2.1. U-Net

U-Net(Ronneberger et al., 2015) was originally established for biomedical segmentation applications. The architecture consists of two main components including a contracting path and a symmetric expanding path. The contracting path can extract the key features of the input, while the expanding path can enable precise localization. In order to localize, the operations (i.e., copying and cropping) linking the first and second paths allow the network to combine high resolution features from the contracting path and the upsampled output from the expanding path so that highly accurate information from the contracting path can be leveraged to generate the segmentation result as close as possible to the ground truth. It has been demonstrated that such a network can be trained end-to-end via few images to achieve precise segmentations. However, the operations of transferring the entire feature map can be computationally expensive, so a new method was developed in SegNet to better build up the connection between two paths without the cost of more memory.

2.2. SegNet

SegNet(Badrinarayanan et al., 2017) has an encoder-decoder architecture, followed by a final pixelwise classification layer. Each encoder performs convolution and max-polling to produce a set of feature maps, while the decoder upsamples its input feature maps via the memorized max-pooling indices from the corresponding feature maps in the encoder. SegNet was first designed as an efficient tool for segmenting road and indoor scene which is efficient in terms of memory and computational time. It is more efficient than U-Net, since it only stores the max-pooling indices of the feature maps and uses them in its decoder network, instead of storing the entire feature maps to capture and store boundary information, and to keep spatial resolution of the feature maps that can be lost upon max-pooling in the encoder network. This new method of upsampling via the memorized max-pooling indices can be also reffered to as *max unpooling*, which has been the main inspiration for *FCDenseNet V2*.

2.3. DenseNet and FCDenseNet

DenseNet(Huang et al., 2017) was first developed on image classification tasks. The key idea of DenseNet is that each layer is directly connected to every other layer in a feed-forward fashion via Dense Blocks (DBs). Each DB is an iterative concatenation of previous feature maps. DenseNet can be seen as an extension of ResNet(He et al., 2016), which performs iterative summation of previous feature maps. However, DenseNet has some interesting features that ResNet does not possess, including parameter efficiency, implicit deep supervision, due to short paths to all feature maps in the architecture, and feature reuse, which means all layers can easily access their preceding layers.

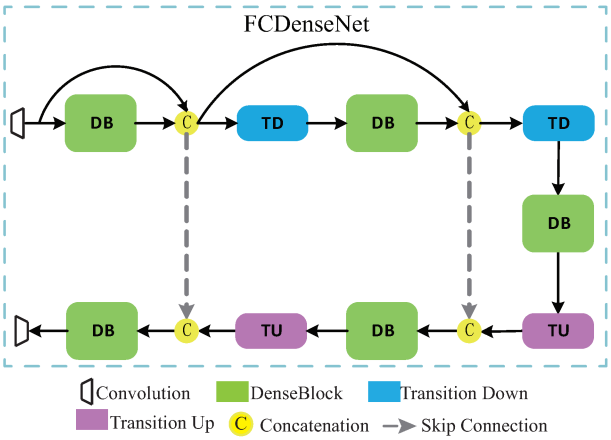

Figure 1. Diagram of FCDenseNet (Duan et al., 2020)

FCDenseNet, which inherits these nice characteristics of DenseNets, can deal with the problem of semantic segmentation, which has achieved good performance on urban scene. The diagram (*Figure* 1) comprises a downsampling path on the top with a 3×3 Convolution, two Transitions Down (TD) and three DBs, and an upsampling path at the bottom with two Transitions Up (TU), two DBs and a 1×1 Convolution(Jégou et al., 2017). Circles stand for concatenation, while arrows represent connectivity patterns. Moreover, gray vertical arrows stand for skip connections, where the feature maps from the downsampling path are concatenated with the counterparts in the upsampling path. In the downsampling path, each DB is followed by a TD, except the last DB, which can be referred to as bottleneck. However, in the upsampling path, each TU is followed by a DB. In the downsampling path, the input of a dense block is concatenated with its output, but it is not in the upsampling path due tot the cost of memory. In the upsampling path, TU consists of a transposed convolution that upsamples the previous feature maps. The upsampled feature maps are then concatenated to the ones coming from the skip connection

to form the input of a new dense block.

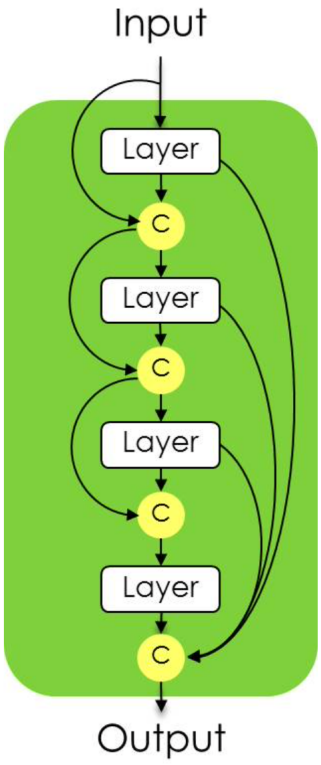

Figure 2. Diagram of Dense Block (DB) (Jégou et al., 2017)

In a downsampling path, the input to a DB is concatenated with its output, leading to a linear growth of the number of feature maps (*Figure* 2). For a 4-layer DB(Jégou et al., 2017), the first layer is applied to the input to create k feature maps, which are concatenated to the input. The second layer is then applied to create another k features maps, which are again concatenated to the previous feature maps. The output of the block is the concatenation of the outputs of the 4 layers, and thus contains $4 \times k$ feature maps. k can also be referred as to growth rate parameter, which is usually set to a small value (e.g. $k = 12$).

Since FCDenseNet has very nice properties and it has not been implemented on semantic segmentation of chest CT images of COVID-19 patients, in this paper, it is proposed as the main framework to segment anomalies in COVID-19 chest CT images, in comparison with FCDenseNet V2, which uses *max unpooling* instead of transposed convolution to reduce the loss of spatial information.

Table 1. Class Sizes

CLASS	PIXEL COUNT ($\times 10^6$)	IMAGE PIXEL COUNT ($\times 10^6$)
BACKGROUND	76.4224	101.1876
GGO	1.2276	42.2052
CONSOLIDATION	0.5689	26.2144
OTHER LUNGS	22.9686	101.1876

3. Methods

3.1. Dataset

Two datasets are obtained from the Kaggle website. One is a dataset of 100 CT image slices with a resolution of 512×512 from more than 40 patients with COVID-19, while the other is a dataset of 829 slices converted and normalized in the same way. All CT images have been manually labeled with four classes, including GGO, consolidation, other lung areas, and background. All data are store in *.npy* format, which can be easily loaded for model fitting. These two datasets are combined as one for all later analyses.

As mentioned above, this combined dataset has 929 CT image slices with a resolution of 512×512, each of which has been manually labeled with four classes. To eliminate any redundant images, all images with more than 90% background are excluded. Furthermore, all images with conflicts in labels of three classes including GGO, consolidation, and background (i.e., same pixel is labelled with different classes) are excluded due to mutual exclusion of classes. Since any pixel that is not labeled as GGO, consolidation, or background should be labeled as other lung areas, the label of other lung areas is updated after all other three classes are confirmed without conflicts.

Only 386 pairs, each of which has a CT frame and the corresponding ground truth, are left after the above exclusion criteria are applied. These pairs can be further used for modelling. Pixel count (total number of pixels in a class) and image pixel count (total number of pixels in images that had an instance of a class) show that it is a very imbalanced dataset (*Table* 1). The most dominant class (i.e., background) is larger in order of 10^2 than the least represented class (i.e., consolidation) in terms of pixel count. Moreover, with respect to image pixel count, only around 26% and around 42% of all images have consolidation and GGO, respectively. Due to the fact that this dataset is highly imbalanced, *median frequency balancing* (Eigen & Fergus, 2015) is applied where the weight assigned to a class in the cross-entropy loss function is the ratio of the median of class frequencies computed on the entire training set divided by the class frequency. More details will be discussed in *Training*.

Figure 3 illustrates the original, the lung-masked, and the labeled images of one sample from these 386 pairs. For modeling, only lung-masked images (*Figure* 3, middle) and their corresponding labels (*Figure* 3, right) are used, so that irrelevant areas in the original CT scans (*Figure* 3, left) can be eliminated at the very beginning.

Figure 4 illustrates the reason why spatial information is of vital importance for semantic segmentation of anomalies. To generate this figure, GGO and consolidation are combined as one class, while the others are grouped as the other class. Thus, a binary mask for GGO and consolidation can be generated for each image. All the binary masks are then summed up to form a graphic that illustrates the regions of the lungs most prune to infection that can lead to GGO and consolidation. Yellow spots are clearly more susceptible to infection compared to other red regions, which demonstrates the importance of spatial information of pixels.

3.2. Model

3.2.1. FCDenseNet

The FCDenseNet used for modeling has the exact architecture as illustrated in Figure 1. It starts with a 2D 3×3 convolution with stride equal to 1 and padding equal to 1. It is then followed by two DB + TDs and one DB as the bottleneck. In the upsampling path, the output from the downsampling path is fed into two TU + DBs, and the network ends with a 2D 1×1 convolution with stride equal to 1 and padding equal to 0, followed by a softmax non-linearity for prediction. Each DB has 4 layers, and each layer creates 12 feature maps. In other words, the growth rate of the layer is set to $k = 12$. Each DB layer is composed of Batch Normalization (BN), followed by ReLU, a same 3×3 convolution and dropout with probability $p = 0.2$. TD is composed of BN, followed by ReLU, a same 1×1 convolution, dropout with $p = 0.2$ and a non-overlapping max pooling of size 2×2. TU is composed of a 3×3 transposed convolution with stride equal to 2 and zero padding. The upsampled feature maps are concatenated to the ones coming from the skip connection which has the feature maps from the downsampling path to form the input of a new dense block in the upsampling path. Since all input images are CT frames, the number of channels of the input is set to one.

3.2.2. FCDenseNet V2

The only difference between the FCDenseNet V2 and the FCDenseNet implemented in this paper lies in TUs. In the FCDenseNet V2, each TU is composed of a 2D 1×1 convolution with stride equal to 1 and zero padding, followed by a max unpooling of size 2×2 that uses the stored pooling indices from the downsampling path, and another same 1×1 convolution. The first 1×1 convolutions helps to keep the same number of feature maps as the output from the corresponding max pooling so that max unpooling can be performed, while the second one helps TU in the FCDenseNet V2 to maintain the same number of output feature maps as TU in the FCDenseNet so that skip connections can be applied.

3.3. Training

3.3.1. 5-fold cross validation

Due to the relatively small dataset, 5-fold cross validation is implemented on the entire dataset (i.e., 386 CT scans and their labels). Thus, each fold has either 77 or 78 CT frames and their labels. In each iteration, one fold is used as a validation set, while the other four folds are used as a training set. Before cross validation is applied, all images are shuffled once to avoid the effect of strong correlations between two consecutive frames.

Before each pair is fed into the network, the CT scan is scaled between 0 and 1 by subtracting the minimum pixel value of the scan and dividing all pixel values by the range (i.e., maximum pixel value - minimum pixel value). For images in the training set only, they are randomly selected with probability equal to 0.5, to receive a rotation with all possible angles and border replication as the method of pixel extrapolation, a cropping with the size of 75% of the original height multiplied by 75% of the original weight using nearest-neighbour interpolation, and/or a horizontal flipping. For images in the validation set, no transformation is applied.

3.3.2. Median Frequency Balancing

Median frequency balancing (Eigen & Fergus, 2015) is performed by only using the training set, due to the highly imbalanced dataset. The weight assigned to a class in the loss function is the ratio of the median of class frequencies divided by the class frequency. Class frequency is calculated based on the number of pixels of class c divided by the total number of pixels in images where c is present. This implies that larger classes in the training set have a weight smaller than 1 and the weights of the smallest classes are the highest.

3.3.3. 3-class segmentation vs. 4-class segmentation

For a 3-class segmentation, GGO and consolidation are grouped as one class, while background and other lungs remain the other two, so that regions of infection can be properly detected. For a 4-class segmentation, background, GGO, consolidation and other lungs remain the four classes as they appear in the dataset.

Figure 3. A sample from the dataset. CT scan (left), masked lungs (middle), and labeled classes (right)

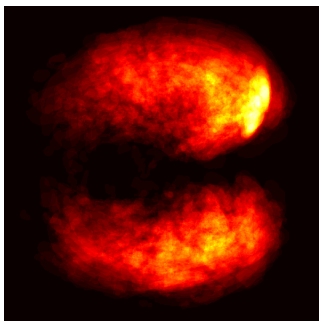

Figure 4. Accumulation of all the labels for GGO and consolidation across the dataset

3.3.4. FCDenseNet vs. FCDenseNet V2

To compare FCDenseNet with FCDenseNet V2 on both 3-class segmentation and 4-class segmentation, four different models (i.e., 3-class FCDenseNet, 3-class FCDenseNet V2, 4-class FCDenseNet and 4-class FCDenseNet V2) are applied on the same training set. During each iteration of the 5-fold cross validation, each model is trained on the same four folds of data and is tested on the same leftover fold.

All models are trained by minimizing the pixel-wise cross-entropy loss with the ADAM(Kingma & Ba, 2014) stochastic optimizer due to its fast convergence rate compared to other optimizers. Median frequency balancing is applied to calculate the weights for the cross-entropy loss function. All models are also regularized with a weight decay of 10^{-4}. The initial learning rate is set to 10^{-4} with an exponential decay of 0.995 after each epoch. Moreover, all models are trained for 100 epochs with batch size equal to 4.

3.4. Testing

As mentioned above, in each iteration of the 5-fold cross validation, each model is trained on the same four folds of data and is tested on the same leftover fold. During training, all models are train for 100 epochs and after each epoch, the validation loss and the corresponding model weights are stored. The model with the minimum validation loss will then be selected for testing.

77 or 78 CT images in one fold are used for testing. A list of evaluation metrics are applied to calculate the accuracy and to assess the performance of each model. For each iteration of the 5-fold cross validation, accuracy is calculated on the image level in the testing fold, and only median accuracy among all testing images is recorded for each model. After the 5-fold cross validation, five medians for each model are stored to compare the performance of different models. Mean and standard deviation of every five medians for each model are calculated to assess the performance and stability of each model. More details about evaluation metrics will be discussed here.

3.4.1. Sensitivity

Sensitivity is defined as true positives divided by the sum of true positives and false negatives for each class c. True positives can be calculated by the number of pixels that are in class c and are classified as class c in an image, while false negatives can be calculated by the number of pixels that are in class c but are not classified as class c in an image.

3.4.2. Specificity

Specificity is defined as true negatives divided by the sum of true negatives and false positives for each class c. True negatives can be calculated by the number of pixels that are not in class c and are not classified as class c in an image, while false positives can be calculated by the number of pixels that are not in class c but are misclassified as class c in an image.

3.4.3. Sorensen-Dice/Dice coefficient

Sorensen-Dice or Dice coefficient is defined as twice the number of true positives divided by the sum of false positives, false negatives and twice the number of true positives.

3.4.4. G-mean

G-mean is defined as the square root of sensitivity multiplied by specificity.

3.4.5. Pixel accuracy

Pixel accuracy is defined as the number of pixels that are correctly classified for all classes divided by the total number of pixels of each image.

3.4.6. Intersection over Union (IoU)

IoU is defined as the number of pixels in the intersection of the prediction and the ground truth for each class in an image, divided by the union.

4. Results

4.1. 3-class Segmentation: FCDenseNet vs. FCDenseNet V2

FCDenseNet V2 is applied on the 3-class segmentation task, in comparison with FCDenseNet. As mentioned above, 3 classes refer to background, infected areas (i.e., GGO + consolidation), and other lungs (i.e., normal areas). The model with the minimum validation loss in each iteration of the 5-fold cross validation is used for testing. Accuracy is calculated on the basis of each image, and median accuracy among all testing images is recorded for each iteration of cross validation. Thus, for each model, five medians are used to calculate the mean and the standard deviation. For this session, two models are compared to each other, including 3-class FCDenseNet and 3-class FCDenseNet V2.

Results are summarized in Table 2 and are visualized in Figure 5. Since areas of infection are of our primary interest, only accuracy for GGO and consolidation combined is shown here. In Table 2, means and standard deviations are reported, and higher means and lower standard deviations between two models are in bold.

Table 2. 3-class Segmentation: Accuracy of GGO and consolidation combined

Metric	FCDenseNet	FCDenseNet V2
Sensitivity	**0.8224** (**0.0591**)	0.7758 (0.1513)
Specificity	0.9832 (**0.0060**)	**0.9842** (0.0077)
Dice coefficient	**0.5106** (0.0929)	0.5101 (**0.0808**)
G-Mean	**0.8888** (**0.0321**)	0.8604 (0.0797)
IOU	**0.3471** (0.0868)	0.3456 (**0.0732**)
Pixel Accuracy	0.9909 (0.0033)	**0.9926** (**0.0026**)

FCDenseNet shows higher mean and median sensitivity with a relatively lower standard deviation, while FCDenseNet V2 shows higher mean and median specificity with a relatively higher standard deviation. While FCDenseNet shows higher means (*Table* 2) in terms of IoU and dice coefficient, FCDenseNet V2 shows higher medians (*Figure* 5). Therefore, FCDenseNet and FCDenseNet V2 have achieved very close performance on the 3-class segmentation task, but FCDenseNet V2 shows a better stability (i.e., lower standard deviations) in terms of IoU and dice coefficient.

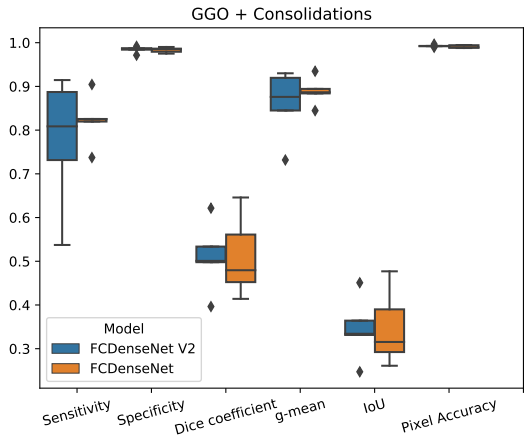

Figure 5. 3-class Segmentation: Accuracy of GGO and consolidation combined

4.2. 4-class Segmentation: FCDenseNet vs. FCDenseNet V2

As mentioned above, 4 classes refer to background, GGO, consolidation, and other lungs. For this session, two models are compared to each other, including 4-class FCDenseNet and 4-class FCDenseNet V2.

Table 3. 4-class Segmentation: Accuracy of GGO

METRIC	FCDENSENET	FCDENSENET V2
SENSITIVITY	0.5200 (0.1496)	**0.5456** (**0.1080**)
SPECIFICITY	**0.9872** (0.0072)	0.9860 (**0.0053**)
DICE COEFFICIENT	**0.3584** (0.0601)	0.3518 (**0.0359**)
G-MEAN	0.7085 (0.1083)	**0.7243** (**0.0784**)
IOU	**0.2198** (0.0451)	0.2140 (**0.0265**)
PIXEL ACCURACY	**0.9914** (0.0029)	0.9909 (**0.0022**)

Table 4. 4-class Segmentation: Accuracy of Consolidation

METRIC	FCDENSENET	FCDENSENET V2
SENSITIVITY	0.7996 (0.1518)	**0.8155** (**0.0744**)
SPECIFICITY	**0.9874** (0.0036)	0.9872 (**0.0035**)
DICE COEFFICIENT	0.5318 (**0.0364**)	**0.5376** (0.0596)
G-MEAN	0.8822 (0.0799)	**0.8948** (**0.0358**)
IOU	0.3632 (**0.0330**)	**0.3696** (0.0548)
PIXEL ACCURACY	**0.9914** (0.0029)	0.9909 (**0.0022**)

4.2.1. GGO

Results for GGO are summarized in Table 3 and are visualized in Figure 6. FCDenseNet V2 shows higher mean and median sensitivity with a relatively lower standard deviation, while FCDenseNet shows higher mean and very close median specificity with a relatively higher standard deviation. While FCDenseNet shows higher means (*Table* 3) in terms of IoU and dice coefficient, FCDenseNet V2 shows higher medians (*Figure* 6).

Therefore, FCDenseNet and FCDenseNet V2 have achieved very close performance for GGO on the 4-class segmentation task, but FCDenseNet V2 shows a better stability (i.e., lower standard deviations) in terms of IoU and dice coefficient. Also, it is very impressive to see that FCDenseNet V2 has achieved lower standard deviations across all evaluation metrics.

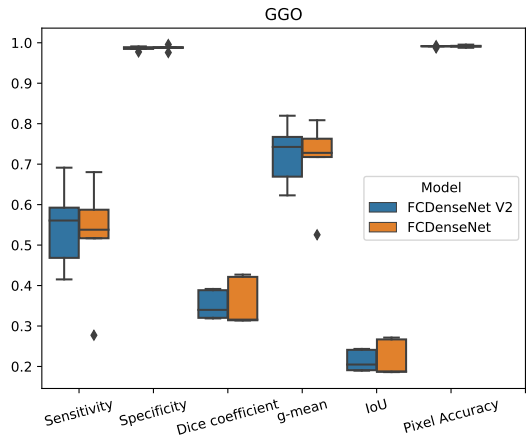

Figure 6. 4-class Segmentation: Accuracy of GGO

4.2.2. CONSOLIDATION

Results for consolidation are summarized in Table 4 and are visualized in Figure 7. FCDenseNet V2 shows higher mean but lower median sensitivity with a relatively lower standard deviation, while FCDenseNet shows higher mean and very close median specificity with a relatively higher standard deviation.

Since FCDenseNet V2 shows higher means (*Table* 4) and higher medians (*Figure* 7) in terms of IoU and dice coefficient, FCDenseNet V2 has achieved better performance for consolidation on the 4-class segmentation task, but FCDenseNet shows a better stability (i.e., lower standard deviations) in terms of IoU and dice coefficient. Also, it is very impressive to see that FCDenseNet V2 has always achieved lower standard deviations or better accuracy or both across all evaluation metrics for consolidation on the 4-class segmentation task.

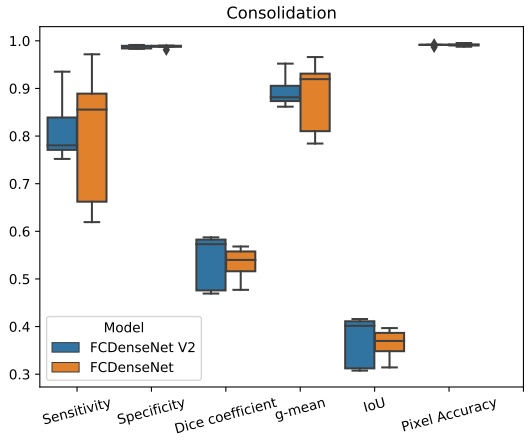

Figure 7. 4-class Segmentation: Accuracy of Consolidation

4.2.3. OTHER LUNGS

Results for other lungs are summarized in Table 5 and are visualized in Figure 8. FCDenseNet shows higher mean but lower median sensitivity with a relatively higher standard deviation, while FCDenseNet V2 shows higher mean and very close median specificity with a relatively higher standard deviation.

FCDenseNet shows higher means (*Table* 5) and higher medians (*Figure* 8) in terms of IoU and dice coefficient. There-

Table 5. 4-class Segmentation: Accuracy of Other Lungs

METRIC	FCDENSENET	FCDENSENET V2
SENSITIVITY	**0.9705** (0.0114)	0.9673 (**0.0099**)
SPECIFICITY	0.9998 (**0.0000**)	**0.9999** (0.0001)
DICE COEFFICIENT	**0.9826** (0.0062)	0.9809 (**0.0049**)
G-MEAN	**0.9840** (0.0062)	0.9823 (**0.0051**)
IOU	**0.9659** (0.0120)	0.9625 (**0.0095**)
PIXEL ACCURACY	**0.9914** (0.0029)	0.9909 (**0.0022**)

fore, FCDenseNet has achieved better performance for other lungs on the 4-class segmentation task, but FCDenseNet V2 shows a better stability (i.e., lower standard deviations) in terms of IoU and dice coefficient. Also, it is very impressive to see that FCDenseNet V2 has achieved lower standard deviations across all evaluation metrics except specificity.

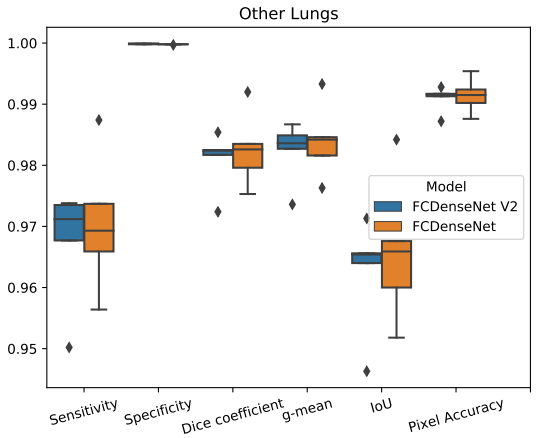

Figure 8. 4-class Segmentation: Accuracy of Other Lungs

4.3. Qualitative Results

More qualitative results can be found in Figure. 9, 10, 11 and 12 in the Appendix. The results with the highest dice coefficient for infection are selected for 3-class segmentation, while the results with the highest dice coefficient for GGO are selected for 4-class segmentation. No difference in performance between FCDenseNet and FCDenseNet V2 can be seen visually in these best results.

5. Summary and Future Work

This work is focused on developing a new approach named FCDenseNet V2 (i.e., FCDenseNet with max unpooling) on sematic segmentation task of COVID-19 CT frames. In comparison of FCDenseNet, FCDenseNet V2 can always show a better performance or a better stability or both on classes in different sizes. When it comes to an extremely small class like consolidation, FCDenseNet V2 tends to have a better performance, while when it comes a larger class like other lungs, it tends to be more stable.

However, more different datasets should be exploited to validate whether FCDenseNet V2 can be generated to other tasks with a consistent improvement in stability or accuracy in comparision with FCDenseNet. Differrent hyperparameter settings should also be investigated, including growth rate, number of DBs on each path, and number of layer of each DB, to further verity whether this improvement is consistent. Other segmentation frameworks should be involved in the similar modification (i.e., using max unpooling) to see whether max unpooling can improve model performance across different architectures.

For this specific task, a highly imbalanced dataset is utilized with median frequency balancing as the main approach to accommodate for the imbalance among different classes. Other techniques should further be investigated to deal with this common issue in image segmentation.

6. Contribution

Joe Liu (80%) wrote the proposal, prepared for the slides, gave the presentation, developed the codebase and wrote the report.

Qifeng Tan (20%) reviewed the proposal, the slides, and the report, and was involved in the discussion for this project.

References

Badrinarayanan, V., Kendall, A., and Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

Caruso, D., Zerunian, M., Polici, M., Pucciarelli, F., Polidori, T., Rucci, C., Guido, G., Bracci, B., De Dominicis, C., and Laghi, A. Chest ct features of covid-19 in rome, italy. *Radiology*, 296(2):E79–E85, 2020.

Chung, M., Bernheim, A., Mei, X., Zhang, N., Huang, M., Zeng, X., Cui, J., Xu, W., Yang, Y., Fayad, Z., et al. Ct imaging features of 2019 novel coronavirus (2019-ncov) radiology. 2020 apr; 295 (1): 202–207. doi: 10.1148/radiol. 2020200230, 2020.

Daimary, D., Bora, M. B., Amitab, K., and Kandar, D. Brain tumor segmentation from mri images using hybrid convolutional neural networks. *Procedia Computer Science*, 167:2419–2428, 2020.

Duan, X., Liu, N., Gou, M., Wang, W., and Qin, C. Steganocnn: Image steganography with generalization ability based on convolutional neural network. *Entropy*, 22(10):1140, 2020.

Eigen, D. and Fergus, R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, pp. 2650–2658, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hope, M. D., Raptis, C. A., and Henry, T. S. Chest computed tomography for detection of coronavirus disease 2019 (covid-19): don't rush the science, 2020.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Jégou, S., Drozdzal, M., Vazquez, D., Romero, A., and Bengio, Y. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 11–19, 2017.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lakshmanaprabu, S., Mohanty, S. N., Shankar, K., Arunkumar, N., and Ramirez, G. Optimal deep learning model for classification of lung cancer on ct images. *Future Generation Computer Systems*, 92:374–382, 2019.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Saood, A. and Hatem, I. Covid-19 lung ct image segmentation using deep learning methods: U-net versus segnet. *BMC Medical Imaging*, 21(1):1–10, 2021.

Wu, F., Zhao, S., Yu, B., Chen, Y.-M., Wang, W., Song, Z.-G., Hu, Y., Tao, Z.-W., Tian, J.-H., Pei, Y.-Y., et al. A new coronavirus associated with human respiratory disease in china. *Nature*, 579(7798):265–269, 2020.

Appendix

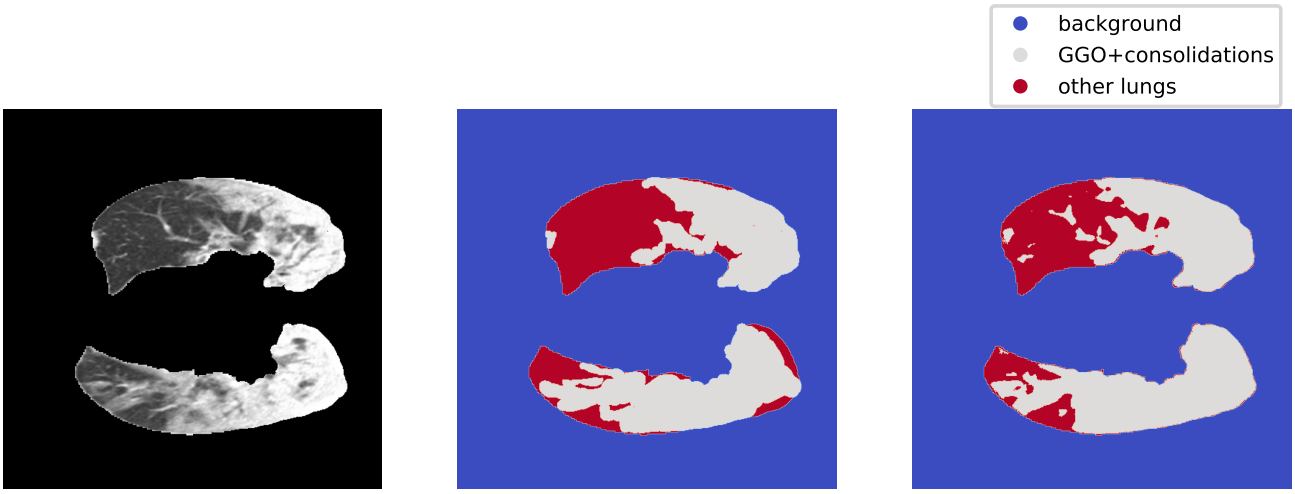

Figure 9. The qualitative result of a sample from 3-class FCdenseNet

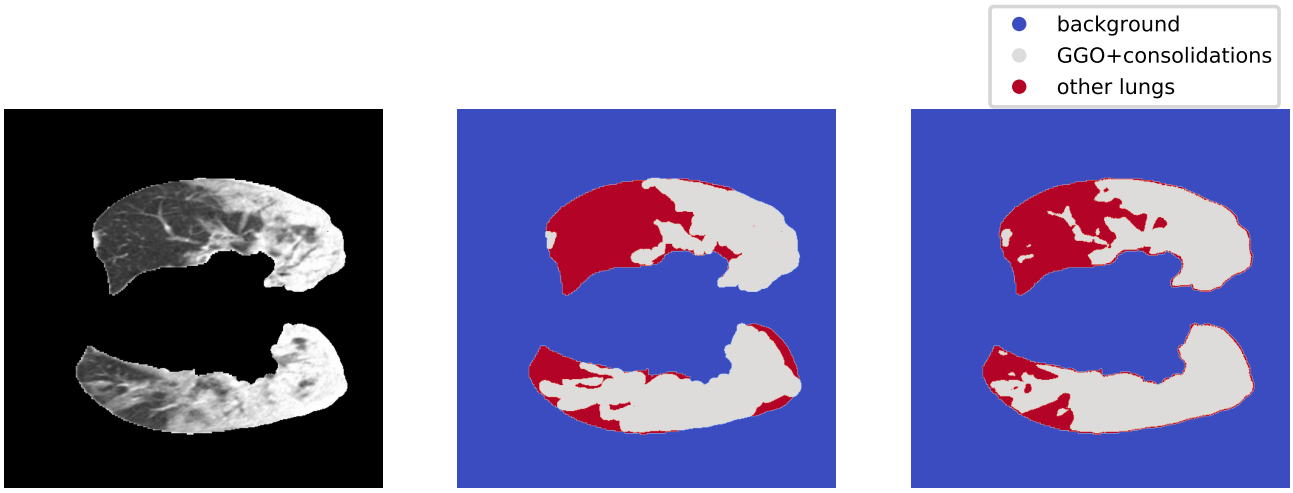

Figure 10. The qualitative result of a sample from 3-class FCdenseNet V2

Figure 11. The qualitative result of a sample from 4-class FCdenseNet

Figure 12. The qualitative result of a sample from 4-class FCdenseNet V2