Received: December 21, 2022. Revised: March 23, 2023.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Segmentation of COVID-19 Chest CT Images Based on SwishUnet

Akhmad Irsyad[1,2] **Handayani Tjandrasa**[1]* **Shintami Chusnul Hidayati**[1]

[1]*Departement of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*
[2]*Information System, Mulawarman University, Samarinda, Indonesia*
* Corresponding author's Email: handatj@its.ac.id

Abstract: Coronavirus disease 2019 (COVID-19), the disease caused by severe acute respiratory syndrome coronavirus-2 (SARS-CoV-2), has been spreading since 2019 until now. Chest CT-scan images have contributed significantly to the prognosis, diagnosis, and detection of complications in COVID-19. Automatic segmentation of COVID-19 infections involving ground-glass opacities and consolidation can assist radiologists in COVID-19 screening, which helps reduce time spent analyzing the infection. In this study, we proposed a novel deep learning network to segment lung damage caused by COVID-19 by utilizing EfficientNet and Resnet as the encoder and a modified U-Net with Swish activation, namely swishUnet, as the decoder. In particular, swishUnet allows the model to deal with smoothness, non-monotonicity, and one-sided boundedness at zero. Three experiments were conducted to evaluate the performance of the proposed architecture on the 100 CT scans and 9 volume CT scans from Italian the society of medical and interventional radiology. The results of the first experiment showed that the best sensitivity was 82.7% using the Resnet+swishUnet method with the Tversky loss function. In the second experiment, the architecture with basic Unet only got a sensitivity of 67.2. But with our proposed method, we can improve to 88.1% by using EfficientNet+SwishUnet. For the third experiment, the best performance sensitivity is Resnet+swishUnet with 79.8%. All models with SwishUnet have the same specificity where the value is 99.8%. From the experiments we conclude that our proposed method with SwishUnet encoder has better performance than the previous method.

Keywords: Segmentation, Resnet, EfficientNet, Unet, SwishUnet, COVID-19.

1. Introduction

COVID-19 is a disease that has become a world pandemic, this disease has spread since 2019 until now. COVID-19 continues to grow and mutate to have several variants such as delta and omicron, this disease originally developed from severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). COVID -19 attacks the lungs of sufferers. COVID-19 sufferers have several characteristics, namely the presence of ground glass opacity (GGO), Consolidation. these characteristics can exist only one or they can appear simultaneously [1].

There are some methods to find out whether someone has positive COVID-19. one of the methods is reverse-transcriptase polymerase chain reaction (RT-PCR), the lack of this method is the process that takes hours, or even days. The second method is rapid test, the rapid test has a faster process but the accuracy is lower than RT-PCT [2, 3]. Therefore, a test that can be done quickly and accurately is needed, one alternative is the use of a CT-scan.

The following is a description of the features of a CT-scan of the lungs of COVID-19 patients. The results of a CT-scan of the lungs of COVID-19 patients have various signs, such as Ground Glass Opacity (GGO), consolidation, and Pleural changes. GGO is the most prevalent imaging finding, with an incidence rate of up to 98%, and is defined as a cloudy area with a slight increase in density in the lungs without blurring of the bronchial and vascular boundaries. This may be caused by partial air displacement caused by partial filling of air spaces or interstitial thickening. The patient, a 35-year-old male who has complained of a fever and a headache for one day, is shown in Fig. 1 (a) as an example of a covid patient with characteristics of ground glass

(a) (b) (c)

Figure. 1 Example CT-scan of a COVID-19 patient with: (a) GGO, (b) Pleural changes, and (c) consolidation [1]

opacity on a CT scan. Right lower lobe ground glass opacity is visible on a CT scan (red frame) [1].

According to a recent study involving 81 individuals with COVID-19, 32 percent of them had pleural changes such as pleural thickening and pleural effusion. CT-scan 80-year-old female COVID-19 patient with a 7-day fever is shown in Fig. 1 (b). CT scan reveals thickening of the left pleura (white arrow) [1].

A rise in lung parenchyma density that conceals the borders of the underlying blood vessels and airway walls is a sign of consolidation, which is defined as the replacement of alveolar air with pathological fluid, cells, or tissue. In Fig. 1 (c), a 47-year-old male COVID-19 patient is shown with fever symptoms that have persisted for seven days. The right lobe subpleural area of the CT scan shows consolidation (red frame) [1].

The CT-scan have fast process and high accuracy, but the increasing number of COVID-19 patients causes radiologists to be overworked and slows down the process of detecting COVID-19 patients. An automatic segmentation system is needed to help radiologists work by reducing the workload and speeding up the screening process so that COVID-19 disease can be reduced. With the development of artificial intelligence (AI) in computer vision, especially in the health sector, AI has been widely used by utilizing CT-scans and X-rays to detect and diagnose pneumonia. However, automatically delineating the area of infection from a chest CT scan is challenging because of the lesions' size, position, and shape in different patients and the low contrast of the infected area on CT images [2].

The research is to modify the Unet architecture which is well known in segmentation. As we know the Unet architecture have two parts, namely encoder and decoder, the encoder function is to extract features from input and decoder to reconstruct the extraction feature to a segmented image. In this research, the Unet architecture is modified to change the encoder to EfficientNet or Resnet because both architectures are known to be used in computer vision and have better results based on some previous research, we modify the decoder by changing the ReLU activation functions with Swish units. The swish activation function has better performance when compared to ReLU, as demonstrated that the InceptionResnetV2 model can increase the classification accuracy by 0.6% on the ImageNet dataset if Swish units replace ReLUs [4]. The proposed architecture is expected to improve the performance of semantic segmentation.

The main objective of this study is to segment COVID-19 infection involving Ground-Glass Opacities (GGO) dan Consolidation areas in chest CT images using encoder part of EfficientNet B0 decoder part of SwishUnet when the other paper only train model only with single label lesion, this research trying a multi-label segmentation COVID-19, This paper proposes to use EfficientNet B0 as an encoder, as it is known that the Unet encoder has a simple form so that the feature extraction process is not optimal [5], using EfficientNet which has a more complex architecture is expected to get better features. extraction. The Unet decoder is modified where the Unet decoder which consists of four layers containing the convolution layer and ReLu activation is converted into the Swish activation function and adds batch and dropout normalization, using the Swish activation function is expected to improve performance. In this study, we compare the loss function to see the impact performance model with different loss functions to get the best loss function for COVID-19 segmentation.

This paper consists of several sections. Section 1 introduces the Covid-19 lesions. Section 2 describes the related work on COVID-19 segmentation. Section 3 describes the proposed methodology used in this study and explains the methods for segmenting COVID-19 lesions using the SwishUnet decoder. Section 4 discusses the experiment and analyses the

results. Section 5 provides conclusions and future research work.

2. Related work

Currently there have been several studies to help detect lung lesion, from several studies using deep learning or machine learning for classification or segmentation, the data used varies, both using X-rays and CT-scans. Here are some studies using machine learning. In the study by Rezaeijo et al. [5], the researchers used feature extraction of gray level co-occurrence matrix (GLCM) features, intensity and histogram features, gray level run length matrix (GLRLM) feature, and trained using random forest (RF), decision tree (DT) and K-nearest neighbour methods (KNN). Researchers used a dataset of 194 patients suspected of COVID-19. In this study, the best performance was using a Random Forest with an F1 size of 96%, a sensitivity of 96%, and a specificity of 99%.

Apart from machine learning, deep learning is used to segmenation. Following are some previous deep learning studies for segmentation. Baheti et al. [6] creates a model using the decoder part of the U-Net segmentation model and the encoder part is EfficientNet B7. Researchers used the India driving dataset (IDD) Lite dataset, which is less than 50MB in size, has a resolution of 320x227, and contains 7 classes. It consists of 1404 training images, 204 validation images, and 408 test images. Which. the best result of this research is UNet with EfficientNetB7 Encoder with a mean Intersection over Union (mIoU) of 0.6276.

Abedalla et al. [7] in this paper makes a comparison between the ResNet50 and EfficientNetB4-Unet models. For the decoder, researchers used two 3 × 3 convolutions, batch normalization (BN) [8], and a ReLU activation layer to reduce overfitting. Data augmentation and dropout regularization are used. The dataset in this paper uses the chest x-ray dataset provided by SIIM-ACR pneumothorax segmentation challenge 2019. The dataset consists of 12,047 training images with 12,954 labels and 3,205 test images. The results of this paper show that EfficientNetB4-UNet has a better performance compared to Resnet-50 with an accuracy of 99.80%, recall of 55.53%, and precision of 69.50%.

Murat et al. [9] conducted research by comparing several U-Net segmentation models. In the encoder section, this research compares the deep learning models VGG16, ResNet101, DenseNet121, InceptionV3, and EfficientNetB5. The decoder in this study uses Unet segmentation. The dataset used in this study uses two datasets. In the first dataset, there are 110 axial CT images of 60 patients infected with COVID-19. In the second data set, 493 images confirmed and segmented by radiologists that there were cases of COVID-19. The ResNet101+Unet model obtained a Dice score of 82.96%, a sensitivity of 80.66%, and a specificity of 99.55%. The EfficientNetB5+Unet model obtained a Dice score of 82.91%, a sensitivity of 77.73%, and a specificity of 99.68% on the COVID-19 segmentation test dataset.

Bizopoulus et al.[10] compared the deep learning (DL) model for lung segmentation and COVID-19 lesions in Computerized Tomography (CT) scanning, which can also be used as a benchmark for testing medical image segmentation models. Unet architecture coupled with pre-initialized and pre-trained encoders (variations VGG, DenseNet, ResNet, ResNext, DPN, MobileNet, Xception, Inception-v4, EfficientNet. COVID-19 data set with 100 CT scan images used for training/validation and combining different data consisting of 829 images from 9 CT scan volumes for testing. The results of this study indicate that efficient B5 has the best performance for segmentation using Unet as a decoder. The results showed the sensitivity was 94.73%,

From some related work we have concluded that CNNs outperform conventional methods because CNN can handle complex tasks, scalable, and more efficient [11]. There is not yet research to improve EffisientNet B0 for COVID-19 segmentation, and they only use the ReLU activation function. In this paper, we propose deep learning for segmentation by replacing ReLUs with Swish units in the decoder part of Unet.

3. Methodology

In this study we have three experimental scenarios using two datasets from the Italian Society of Medical and Interventional Radiology as follows:

- Train the first model using 54 COVID-19 CT images from the first dataset for GGO and Consolidation, and the background.
- Train the second model using 99 COVID-19 CT images from the first dataset for all lesion areas and the background.
- Train the third model using 373 COVID-19 CT images from the second dataset for all lesion area and the background.

The research process begins by selecting the dataset, after that pre-processing the dataset to get the image improvement, the next process is to build the architecture to train the dataset, we train our proposed

method and for compare we train modified EfficientNet with Unet basic decoder [10], Resnet with Unet basic decoder [7], EfficientNet with SwishUnet, Resnet with SwishUnet. The final process is evaluating the model semantic segmentation

3.1 Pre-processing

The initial step before the dataset is fed into the deep learning model is known as pre-processing. It is necessary to resize the CT-scan because the lungs are very large. The data has been scaled to 512x512, the next process is removing the noise with contrast limited adaptive histogram equalization (CLAHE) in the image so we can see the lesion more detail, The CLAHE outcome of a lung CT picture is shown in Fig. 3. After that the dataset are augmented to increasing variety the dataset.

In CLAHE, the image is split into roughly identically sized non-overlapping regions. Each region has its histogram calculated. The height of each histogram is then redistributed so as not to surpass the limit at which it was clipped. For grayscale mapping, the cumulative distribution function (CDF) is then computed. By combining linearly, the four nearest region mappings, the pixels are finally mapped. The procedures for CLAHE are [12]:

Input: image l
1. Separate the original image into smaller sub-images.
2. For each sub-image in image, do:
   1. Determine the histogram for each sub-image
   2. Clip Limit
      $N_{CL} = N_{CLIP} \times N_{avg}$
      // $N_{avg} = \frac{N_X \times N_Y}{N_{gray}}$
      //N_{gray}=number of grey levels in sub-image
      // N_X , N_Y= the sub-pixel image counts in the X and Y dimensions
      // N_{CLIP} is the sub-highest average pixel value of the image for each grayscale
   3. Clipping the Histogram every using N_{CL}
      // *the grey levels* > N_{CL} the grey levels will be cropped.
   4. Distribution over remaining pixels
      $N_d = \frac{N_{TC}}{N_{gray}}$.
      //the total number of cut-off pixels (N_{TC})
      // the number of pixels (N_d) which are equally adjusted to each greyscale
End.
3. Combine the sub-images
Output: CLAHE processed Image input

3.2 Proposed architecture

In this study, we use Unet as a base architecture. The Unet have two parts, namely encoder and decoder [13], the encoder extracts feature from the source image, to capture high-level image details throughout this procedure, the resolution and feature maps of the image samples are reduced. Modern convolutional neural networks like Resnet [14], MobileNet [15], EfficientNet [16], and InceptionResnetV2 [17] are used as encoder. In the encoder, the input data is processed to get the final feature map. The next process is the encoder feature map which is enhanced by the decoder module layer to recover spatial information. The Unet process gradually lowers the resolution of the input image, It is a challenge to reconstruct the segmentation map to the original image size from the reduced feature map.

The study proposed modified EfficientNet architectures as encoder. Baheti et al. [6] make architecture with EfficientNet and Unet, in our proposed architecture we modified the last layer we only use sixth layer, the purpose is to improve training time but keep the performance good, Beside the encoder, we changed the standard block of the Unet decoder in our proposed model. More detail architecture is shown in Fig. 6.

The fundamental component of the EfficientNet design is the mobile inverted bottleneck convolution (MBConv) with the squeeze and excitation optimization. The number of MBConv blocks varies across the EfficientNet network. The model's depth, width, resolution, and size continue to increase as it transitions from EfficientNetB0 to EfficientNetB7, and its accuracy also rises [6].

Unet basic decoder consists of a convolution layer and ReLU activation as can be seen in Fig. 5, while the decoder in our proposed model uses a convolution layer, swish activation, batch normalization and dropout, for more detail the proposed decoder can be seen in Fig. 4. swish unit is a new activation function that previous research shows it is better than ReLU. Using batch normalization and dropout makes the model learning process more stable.

In the decoder process. the information obtained from efficientNet is upsampled, next is the concat process, the concat process is carried out by combining information from the encoding and decoding processes. while [6] use full architecture of EfficientNet, we modified to make the training proses more faster by only suing six layer and the difference with Baheti et al. [6] architecture is the location of the skip connection, for more details, see Fig. 6.

The architecture we propose for the encoder is EfficientNetB0. In this study we use an input size of

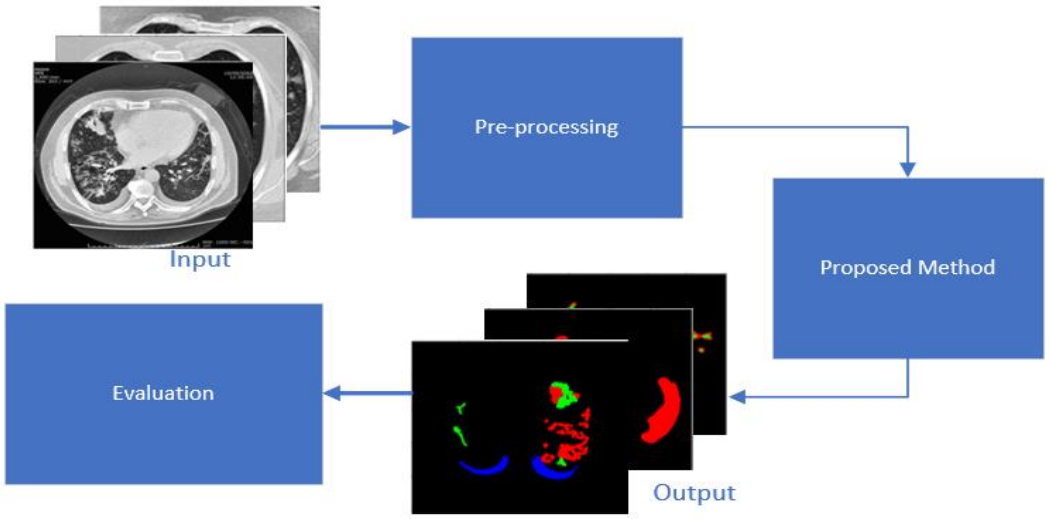


Figure 2. The research process

512 x 512. EfficientNetB0 consists of 6 blocks, the first blocks consist of 1 stack MBconv, the second and third blocks each 2 MBconv, and the fourth and fifth blocks each 3 MBconv, while the sixth block has 1 MBconv. The last block of EfficientNet B0 proceeds to the decoder process using the proposed Unet block. The decoder that we proposed also tested with encoder Resnet [18].

3.2.1. Rectified liner units (ReLU)

An activation function was introduced by [19], known as Rectified Linear Units (ReLU). Deep neural network training is enhanced with ReLU. One of the most well-liked activation mechanisms is ReLU. [20]. ReLU essentially merely establishes a limit on values at zero, such that if x<0 then x = 0 and if x > 0 then x = x [21].

$$f\ (x)\ =\ max(x). \qquad (2)$$

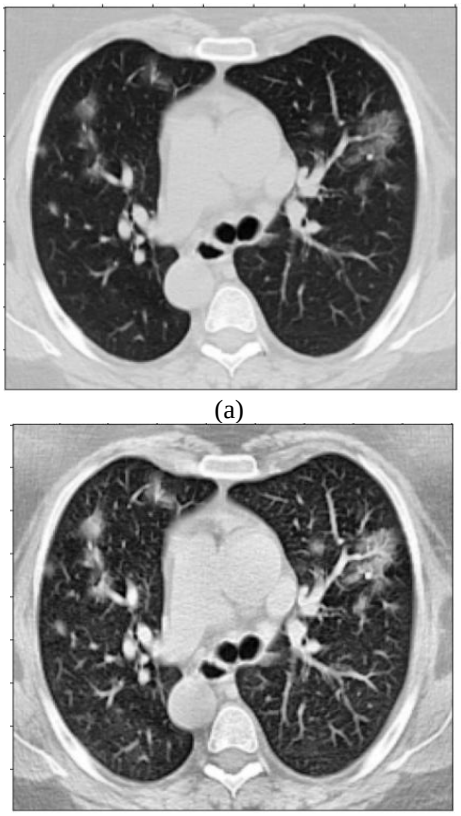
3.2.2. Swish

Swish is a deep learning activation system developed by the Google Brain team. Swish possesses the characteristics of smoothness, non-monotonicity, and one-sided boundedness at zero. They demonstrated that the InceptionResNetV2 model can obtain 0.6 percent more accuracy on the ImageNet dataset if Swish replaces ReLU [4].

$$Nf(x)\ =\ x \cdot sigmoid(x). \qquad (3)$$

3.3 Loss function

The deep learning training model is optimized using the loss function. The loss function's objective



(a)

(b)

Figure. 3 Image CT scan: (a) before preprocessing and (b) after preprocessing

is to reduce error. Loss functions play an essential role in determining the model performance, highly imbalanced segmentation works better with focus based loss functions [22]. The segmentation of the model is improved by a lower value loss function. Following are some loss function techniques:

A well-known technique for determining the loss function in deep learning is binary cross-entropy,

which produces a value between 0 and 1. Binary cross-entropy is described as [23]:

$$L(\mathrm{y},\hat{y}) = -(\,\mathrm{y}\log(\,\hat{y}\,) + (1-y)\log(1-\,\hat{y})) \quad (4)$$

Where $\hat{y}$ is predicted probabilities, and y is groundtruth of data. The cross-entropy loss has several variant methods, one of the variants is softmax cross-entropy where the predicted probabilities ($\hat{y}$) use softmax [24]. Softmax cross-entropy is described as:

$$L(\mathrm{y},\hat{y}) \;=\; -\tfrac{1}{N}\textstyle\sum_{i=1}^{N} y_i^T \log(\hat{y}_i). \quad (5)$$

A variation of cross-entropy is weighted cross-entropy (WCE). In this instance, coefficients are used to weigh the value from the prediction. WCE is frequently employed when dealing with imbalanced data [25]. The definition of the weighted cross-entropy is:

$$L_W(\mathrm{y},\hat{y}) = -(\,\beta * \mathrm{y}\log(\,\hat{y}\,) + (1-y)\log(1-\,\hat{y})) \quad (6)$$

False positives and false negatives can be adjusted using the β parameter. For instance, set β > 1 to lower the number of false negatives, and β<1 to raise the number of false positives [25]. A common metric in the field of computer vision for determining how similar two images are called the dice coefficient. Next, it was modified as Dice Loss, a loss function, in 2016 [25]. Where 1 is added to the numerator and denominator to make sure that the function is not undefined in extreme cases like when $y = \hat{p} = 0$ [25].

$$DL(\mathrm{y},\hat{y}) = 1 - \frac{2y\hat{p}+1}{y+\hat{p}+1}. \quad (7)$$

The Tversky loss, also referred to as a generalization of the Dice coefficient with the aid of the coefficient, it gives false positives and false negatives more weight.

$$DL(\mathrm{y},\hat{p}) = 1 - \frac{y\hat{p}}{y\hat{p}+(\,\beta*\mathrm{y}\log(\,\hat{y}\,)+(1-y)\log(1-\hat{y}))}. \quad (8)$$

3.4 Evaluation metrics

Accuracy can be defined as representing the percentage of image pixels that are correctly classified. In segmentation it is also known as overall pixel accuracy. This is the most basic performance metric but has the limitation of describing the performance of erroneous image segmentation in the event of a class imbalance. Class imbalance occurs

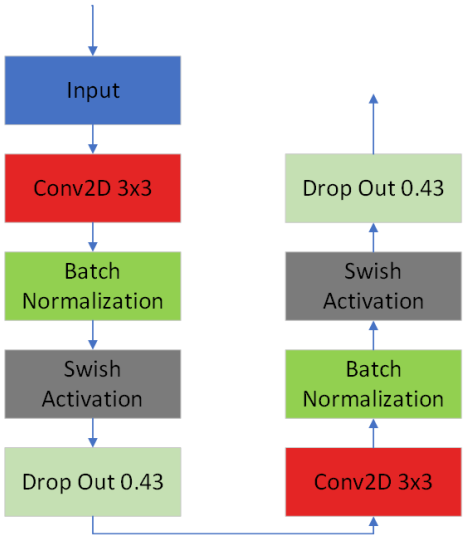


Figure. 4 Proposed decoder

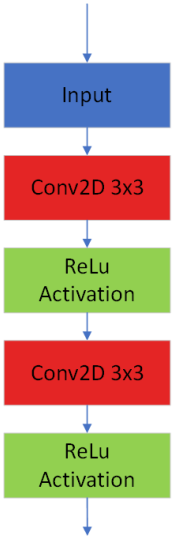


Figure. 5 Plain decoder unit in UNet [26]

when one segmentation class dominates another. In such a case, the higher accuracy for the dominating Class will mask the lower accuracy associated with the other classes thereby giving biased results. Therefore, the accuracy measure is recommended to be used to evaluate segmentation performance with images when there is no class imbalance [27]. The accuracy is described in Eq. (9), as:

$$Accuracy \;=\; \frac{TP+TN}{TP+FP+FN+TN} \quad (9)$$

The true positive and false negative values are used to measure the sensitivity of the model for the segmented area which is defined as:

$$Sensitivity = \frac{TP}{TP+FN} \quad (10)$$

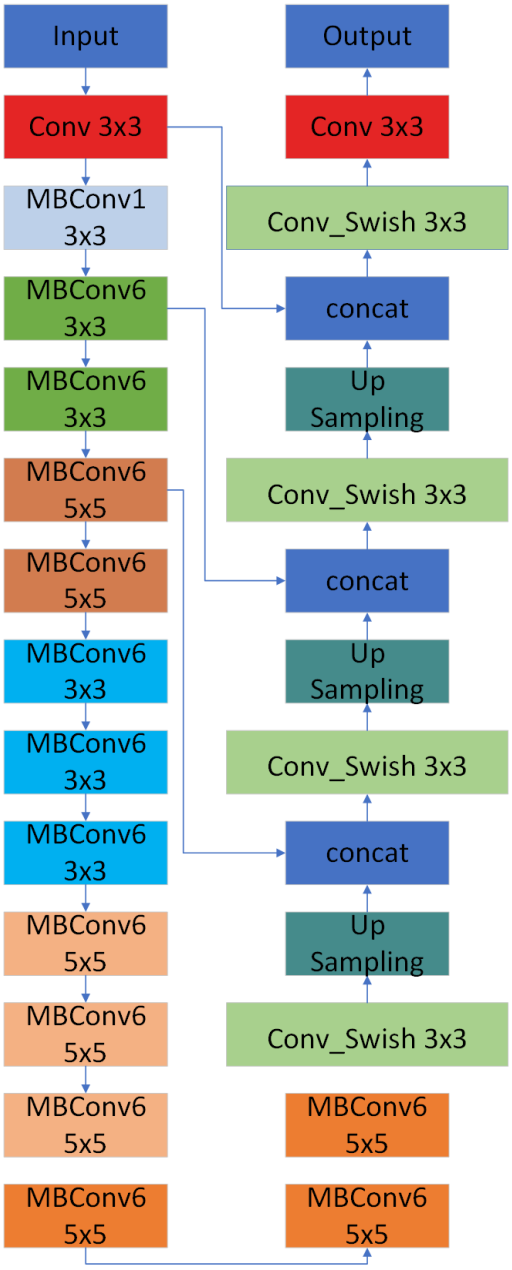


Figure. 6 Proposed architecture model

The true negative and false positive values are used to measure the specificity of the model as given by:

$$Specificity = \frac{TN}{TN+FP} \quad (11)$$

Where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

4 Experiment

This study was carried out in workstation with a Core i7 (8th Gen) 2.20 GHz CPU with 8GB RAM; 6GB graphic memory and implemented in library TensorFlow Python. The study began with collecting images from medical segmentation datasets [8]. After that, the images were pre-processed by resizing to 512x512 and enhancing using CLAHE. The next process was to train the data using deep learning. In this study we compared our proposed method with the architecture Resnet+Unet [7] and EfficientNet+Unet [10].

Researchers conducted three experiments using two datasets. The first experiment used only 54 CT scans that had lesion consolidation and GGO. The first experiment only used GGO and Consolidation lesions because Plural change lesions were only on 25 CT-scans. due to a lack of datasets with plural change causing the plural change to be undetectable. The second experiment used 99 CT-scan. In second experiment, we combined the three lesions into one lesion. the third experiment used a nine volume CT-scan dataset, from this dataset we obtained 373 CT scans of COVID-19. For each dataset, we split 80% of the data for the training data and 20% for the test data. In this paper we explain the three experiments and the experimental results.

4.1 Dataset

In this research, we make use of two datasets from the Italian society of medical and interventional radiology [28], The first dataset consists of 99 COVID-19 CT-scans that are provided with ground truth lesions. doctors from several hospitals established the ground truths. There are three labels for lesions: pleural changes, consolidation, and ground glass. A nine-volume COVID-19 CT scan dataset makes up the second dataset. The datasets are compiled to NIFTI format. The size of dataset is 512x512 pixels, an example dataset and ground truth are shown in Fig. 7. In the ground truth image, red indicates ground glass opacity, green indicates consolidation, and blue indicates pleural changes.
The first datasets have some types of labels, the first type, the dataset only has one lesion, the example is a CT-scan in Fig. 7 (d) and ground truth in Fig. 7 (a), in the ground truth image we only see a red lesion, it means ground truth only has GGO. The second type ground truth is the CT-scan with two lesions, the example is a CT-scan in Fig. 7 (e) and the ground truth in Fig. 7 (b), as we can see the ground truth has two color lesions with GGO and consolidation. The third type is a dataset with three lesions. The CT-scans with three lesions only have 25. The most CT-scans we have is CT-scan with GGO and consolidation with 54 CT-scans, and we have 20 CT-scans with only one lesion. Specifically, the one lesion dataset only has GGO, so in total the first dataset has 99 CT-scans.

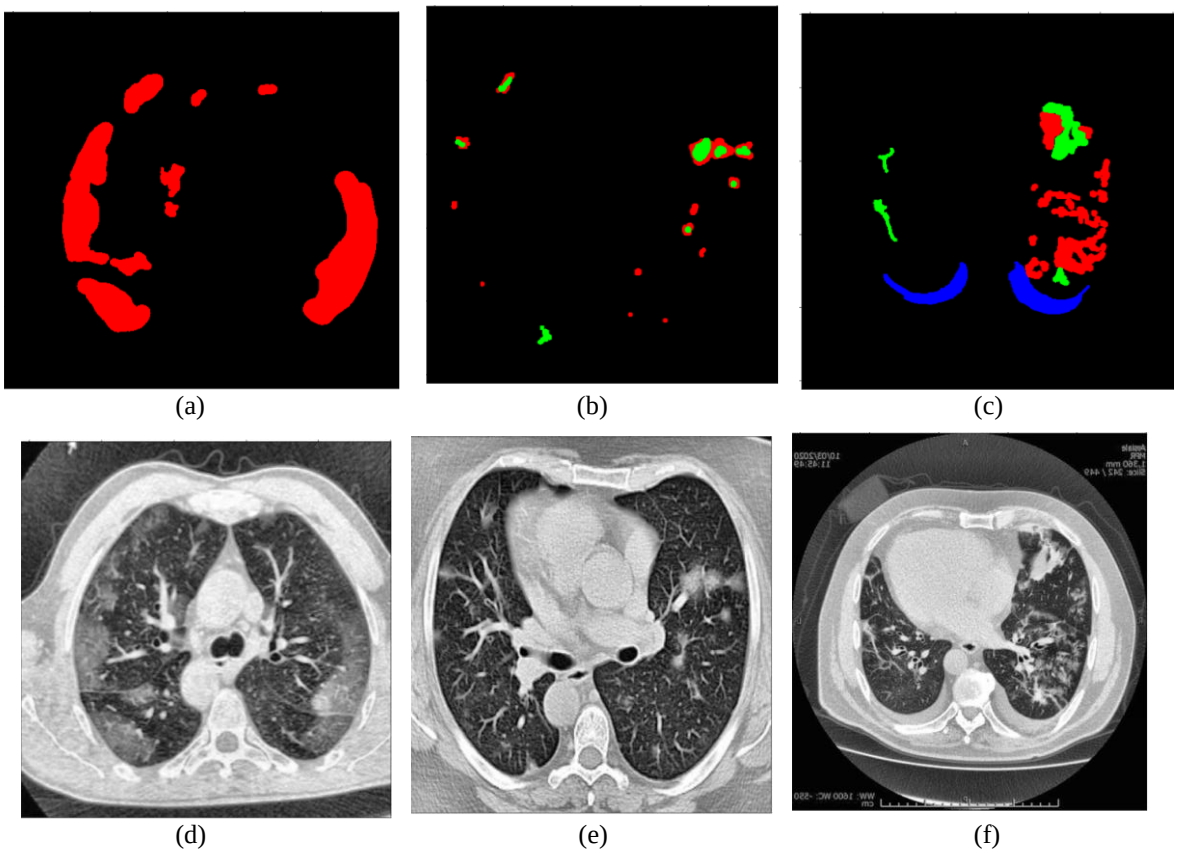


Figure. 7 : (a) Ground truth with one lesion, (b) Ground truth with two lesion, (c) Ground truth with three lesion, (d) CT-scan patient COVID-19 with GGO lesion , (e) CT-scan patient COVID-19 with GGO and consolidation lesion, and (f) CT-scan patient COVID-19 with GGO, consolidation, plural change [28]

The second dataset consists of nine volume CT-scans, the total CT-scans in second dataset is 830 slice CT-scans, but only 373 CT-scans that have lesion. The second dataset have GGO and consolidation, the lesion in the second dataset is combined to be one lesion. The reason we combine the lesions is because in the third experiment we try to test our model if the lesions is one type.

4.2 Result of proposed method

The result of first experiment with using our proposed method can be seen in Table 1, we used EfficientNet and resnet as encoders and we compared loss functions of Softmax cross-entropy, weighted softmax cross-entropy, dice loss, and Tversky. The first experiment use CT-scans with lesion GGO and consolidation. For example, in Fig. 9 the ground truth has mostly GGO (red) and small lesion of consolidation (green). The output of the proposed model is shown in Fig. 8.

In the first experiment, the best sensitivity of GGO lesions is 82.7% by using Resnet+swishUnet with Tversky loss, and the output for consolidation lesions has better performance using Resnet+swishUnet with SoftMax cross-entropy weighted with a score of 73,5%. The best specificity for GGO is 98.9% with EfficientNet+swishUnet and the loss dice loss function, Resnet+Unet using the SoftMax cross-entropy loss function is the best for consolidation specificity with a value of 99.1%. When we average by architecture, we get the best performance is EfficientNet+swishUnet which has an average sensitivity of 51.1% and an average specificity of 96.7%. If the performance in the first experiment is grouped by loss of function, then the softmax cross-entropy is the best result sensitivity with 63.4%. Based on this result we try if three lesion we have marge to be one lesion, and for the second experiment we only use loss function cross-entropy and dice loss.

Fig. 8 shows some models with proposed architecture have big lesion consolidation (symbolized by green), the reason the model assumes there is Consolidation in the image is the similarity of GGO pixels to the lungs. In addition, the mixing of the two labels can cause the model to incorrectly predict the segmentation results. Based on this result we try if three lesion we have marge to be one lesion, but for the second experiment we only use loss function binary cross-entropy and dice loss, because this two is the most used based on literature review.

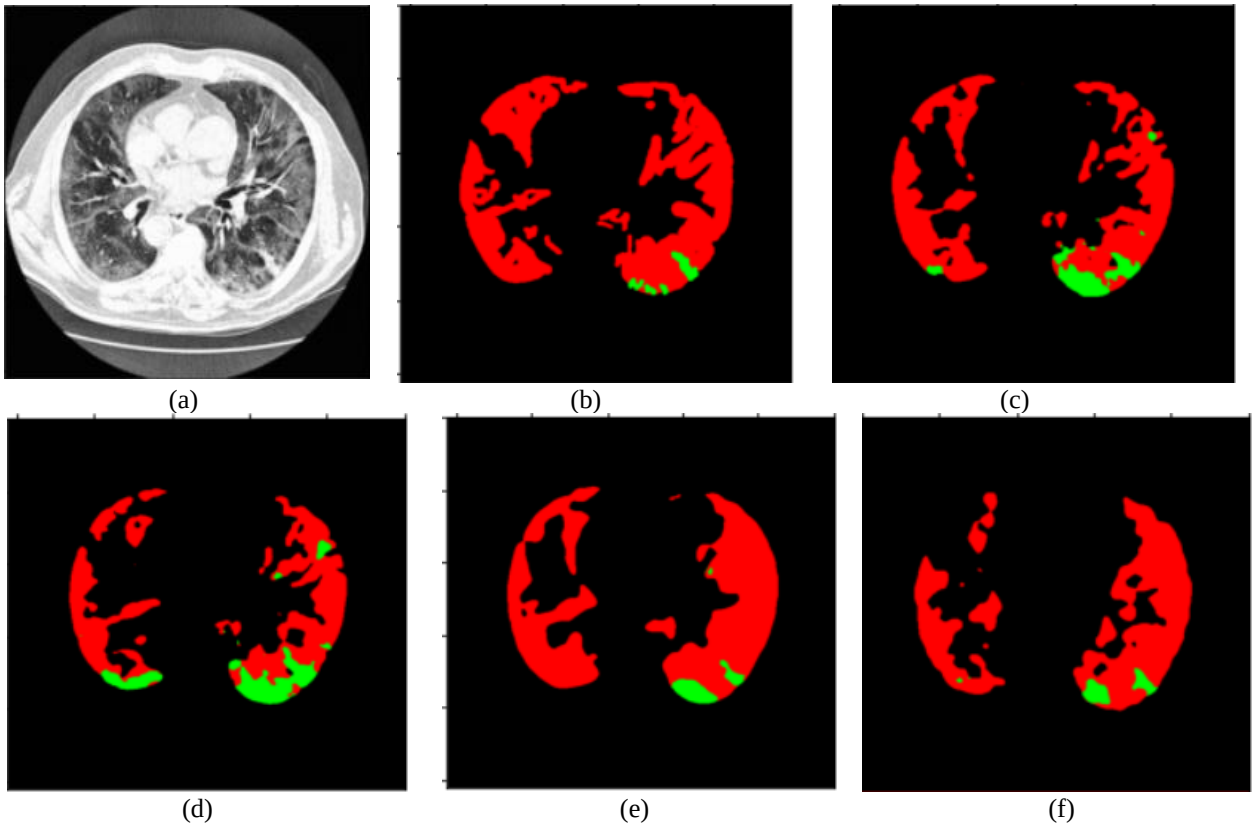


Figure. 9: (a) image test CT scan COVID-19, (b) Groundtruth image (c) Resnet+Unet, (d) EfficientNet+Unet, (e) Resnet+SwishUnet, and (f) EfficientNet+SwishUnet

Table 1. Performance in the first experiment

Model	Loss Function	Acc(%,)	Sens1(%,)	Spec1(%,)	Sens2(%,)	Spec2(%,)
EfficientNet+ swishUnet	Softmax cross-entropy	90.1	68.7	94.1	52.0	97.1
	Dice loss	90.1	46.0	98.9	43.9	98.6
	Softmax cross-entropy weighted	88.3	80.3	91.7	36.0	96.9
	Tversky loss	93.1	49.9	97.8	31.7	98.3
Resnet+ swishUnet	Softmax cross-entropy	93.4	79.8	95.4	58.0	**99.1**
	Dice loss	91.6	70.3	95.6	49.4	97.3
	Softmax cross-entropy weighted	92.9	68.5	97.9	**73.5**	95.9
	Tversky loss	93.1	**82.7**	95.0	66.9	98.9
EfficientNet+ Unet [10]	softmax cross-entropy	94.8	67.0	97.4	65	99.2
	Dice loss	92.2	0	**99.8**	7.8	98.8
	Softmax cross-entropy weighted	92.8	44.8	97.9	44.8	97.8
	Tversky loss	82.8	68.3	86.2	61.1	96.8
Resnet+Unet [7]	Softmax cross_entropy	**94.7**	61.1	98.0	55.6	98.9
	dice_loss	93.5	45.4	98.6	60.6	96.9
	Softmax cross-entropy weighted	94.5	66.3	98.0	64.6	98.2
	Tversky loss	93.5	57.0	97.2	66.9	98.9

The dataset for the second experiment is using dataset 54 CT scans that had lesion consolidation and GGO, but in this experiment, we combine these two labels to be one. The result of the second experiment can be seen in Table 2, the model EfficientNet +SwishUnet with dice loss has the best accuracy where the value is 96.8%. The best performance sensitivity with our proposed model is Efficient+swishUnet with 87.7% but it is little bit lower than Efficient+Unet. The best Specificity with our proposed model is Resnet+swishUnet with loss

Table 2. Performance proposed method in second experiment

Model	Loss Function	Accuracy (%)	Sensitivity (%)	Specificity (%)
EfficientNet+swishUnet	binary crossentropy	96.6	87.7	97.0
EfficientNet+swishUnet	Dice loss	**96.8**	80.3	97.7
EfficientNet+Unet [10]	binary crossentropy	96.7	85.4	97.2
EfficientNet+Unet [10]	Dice loss	96.5	**88.1**	96.9
Resnet+swishUnet	binary crossentropy	96.6	76.4	97.8
Resnet+swishUnet	Dice loss	96.3	70.1	**98.2**
Resnet+Unet [7]	binary crossentropy	**96.8**	79.9	97.7
Resnet+Unet [7]	Dice loss	96.7	80.0	97.6
Unet [29]	Dice loss	-	67.2	96.0

function binary cross-entropy where the value is 97.0%.

The third experiment use nine volumes CT-scan where we get 373 CT-scan. Because from total 830 images only 373 have lesion in the ground truth, the label ground truth is considered by 0 for background and 1 for lesion. In third experiment, we train model with loss function binary cross-entropy. The CT-scan test data, the ground truth and the output of every model that test in third experiment can be seen in Fig. 11.

The result of third experiment can be seen in table 3, the EfficientNetB0+swishUnet have the lowest accuracy with 99.3% but other models have same accuracy where the value is 99.5%, the best performance sensitivity is Resnet+swishUnet with 79.8%. All models have same specificity where the value is 99.8%. The output of third experiment shows in Fig. 13. As we can see, all outputs detect the CT-scan have lesion in the left lung, but in the ground truth in Fig. 12 (b) there is none. If we compare the output The model EfficientNetB2 +SwishUnet have the smallest noise but also have the smallest lesion.

4.3 Result of the basic Unet decoder and comparison

In the first experiment, For the basic Unet architecture we use Resnet+Unet [7], and EfficientNet+Unet [10], beside architecture we compare loss function to see if loss function has effect in training. Table 1 is the performance of Resnet+Unet and EfficientNet+Unet. From the result from the architecture with Unet, we can see that efficientNet+Unet is better in detection GGO where they have sensitivity of 68.3% and specificity of 99.8%. but for consolidation the best performance decoder is Resnet+SwishUnet with sensitivity of 66.9%.

When the performances in the first experiment are grouped by the architecture, we will get the best performance in basic architecture is Resnet+Unet. The average EfficientNet+Unet sensitivity is 44.9%, specificity is 96.8%, and accuracy is 90.6%, and the average Resnet+Unet sensitivity is 59.7%, specificity is 98.1%, and accuracy is 94.1%. The experiment shows the proposed architecture have better performance if compared to segmentation with Unet.

The architectural with basic Unet decoder output in the first experiment shows that the consolidation of the lesion appears to be excessive, where the lesion appears larger than the ground-truth image, but for GGO lesion the output of model look smaller than ground-truth, event in EfficientNet+Unet with loss function Dice Loss the GGO lesion not detection. But the performance of the proposed model still batters if compared to basic architecture.

In the second experiment for basic architecture, we use Resnet+Unet and EfficientNet+Unet with binary cross-entropy and dice loss functions. The reason we only use these two loss functions because binary cross-entropy and dice loss have been used mostly in our literature review. The result of basic Unet decoder model is showed in Table 2. From the result with basic architectures, we can see the EfficientNet+Unet has the best performance with sensitivity 88.1%, the best performance we get by using dice loss as loss function. For Specificity and accuracy, the best performance is EfficientNet +Unet with loss function binary cross-entropy with 97.8% and 79.9%.

The Resnet+Unet inn second ecperiment have the best average performance with sensitivity 87.2%, that means the EfficientNet+Unet more stable even the loss function is change if compared to Resnet+Unet. Our proposed method has better performance than previous method like Resnet+Unet and Unet. For showing the comparison of output model and the ground truth, we can see in Fig. 10 is CT-scan and the ground truth, for the output model we can see in Fig. 11. Our proposed method with SwishUnet encoder has better performance than previous method.

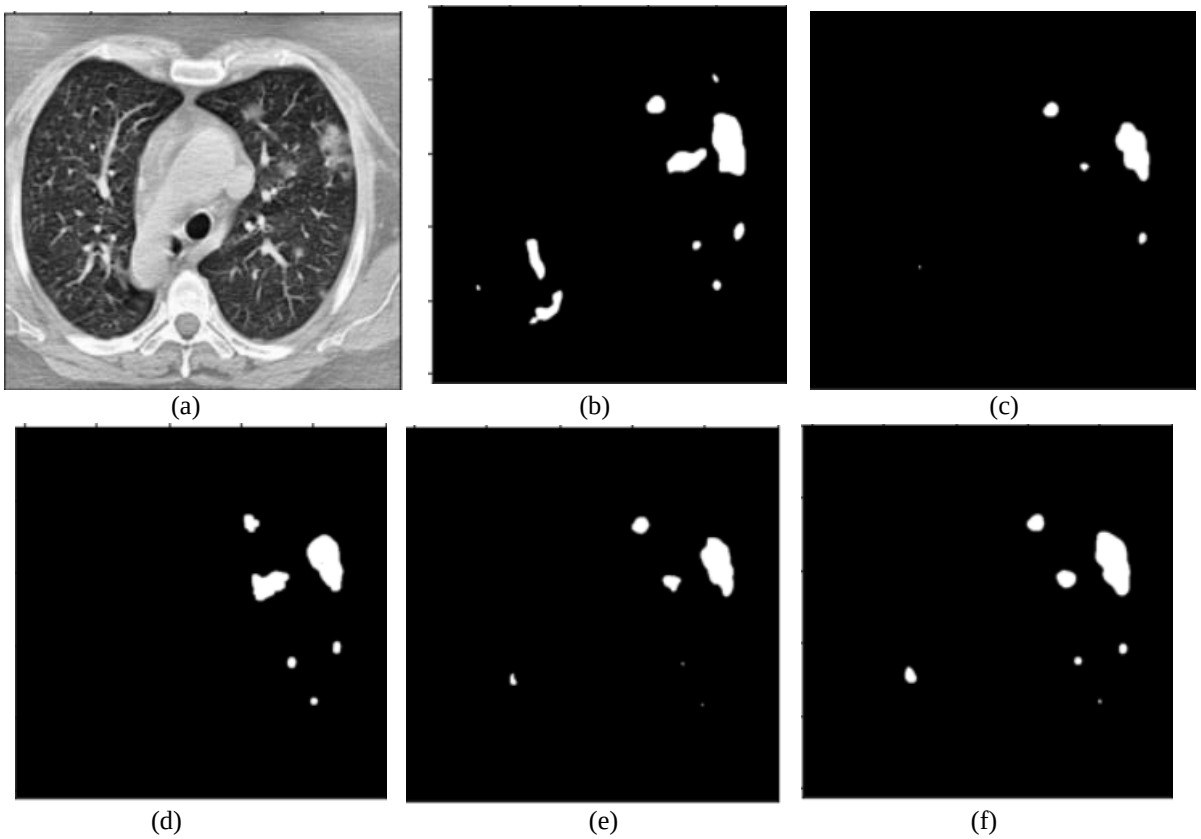


Figure. 10: (a) CT-scan data test second experiment, (b) ground truth data test. (c) Resnet+Unet with loss function binary cross-entropy, (d) EfficientNet+ Unet with loss function binary cross-entropy, (e) Resnet+ Unet with loss function dice loss, and (f) EfficientNet+ Unet with loss function dice loss

The output in Fig. 11 shows the model with loss function binary cross-entropy have some false positive because in the ground truth the in lesion only in the right lung, but the output models have some little lesion in left lung that is a reason why model with binary loss function binary cross-entropy have better Sensitivity, because the model more sensitive if compared to model with dice loss. the output from model with Unet have bigger lesion if compared to model with SwishUnet that's make the sensitivity with Unet little bit higher

If we compare first and second experiment, in the first experiment we have more information because we can see the GGO and consolidation lesion, in second experiment we only know if lung CT-scan have lesion, but we do not know the type of lesion. But if we see performance, the second experiment have better result. Based on compared, In the third experiment researcher trying new dataset with the method in second experiment where the lesion is merged, and we use binary cross-entropy for loss function.

The result of the third experiment can be seen in Table 3, the basic architectures have the same accuracy and specificity where the value is 99.5% and 99.8%. but for sensitivity, the best performance is shown by Resnet+Unet with a value of 77.8%. for Efficient+Unet sensitivity is 72.1%. The output of the third experiment shows in Fig. 17. As we can see, all outputs detect the CT-scan have a lesion in the left lung, this output showed in Fig. 11 (a) and (c), but in the ground truth in Fig. 10 (b) there is none.

The output model Resnet+Unet and Efficient+Unet not only have the smallest noise but also have a false lesion. The output of the base method identifies some normal pixels as lesions but are not actually lesions. This can occur due to several factors, such as insufficient dataset quantity or poor dataset quality so it is difficult for the model to detect lesions due to the similarity of pixel colors between lesions and normal pixels.

From our experiments, we know that our proposed model for segmentation have better performance if compared to basic Unet decoder architectures, but proposed method can be improved with using attention gate (AG) [30] because it still lacking in detecting the small lesion and still false detected some noise as lesion.

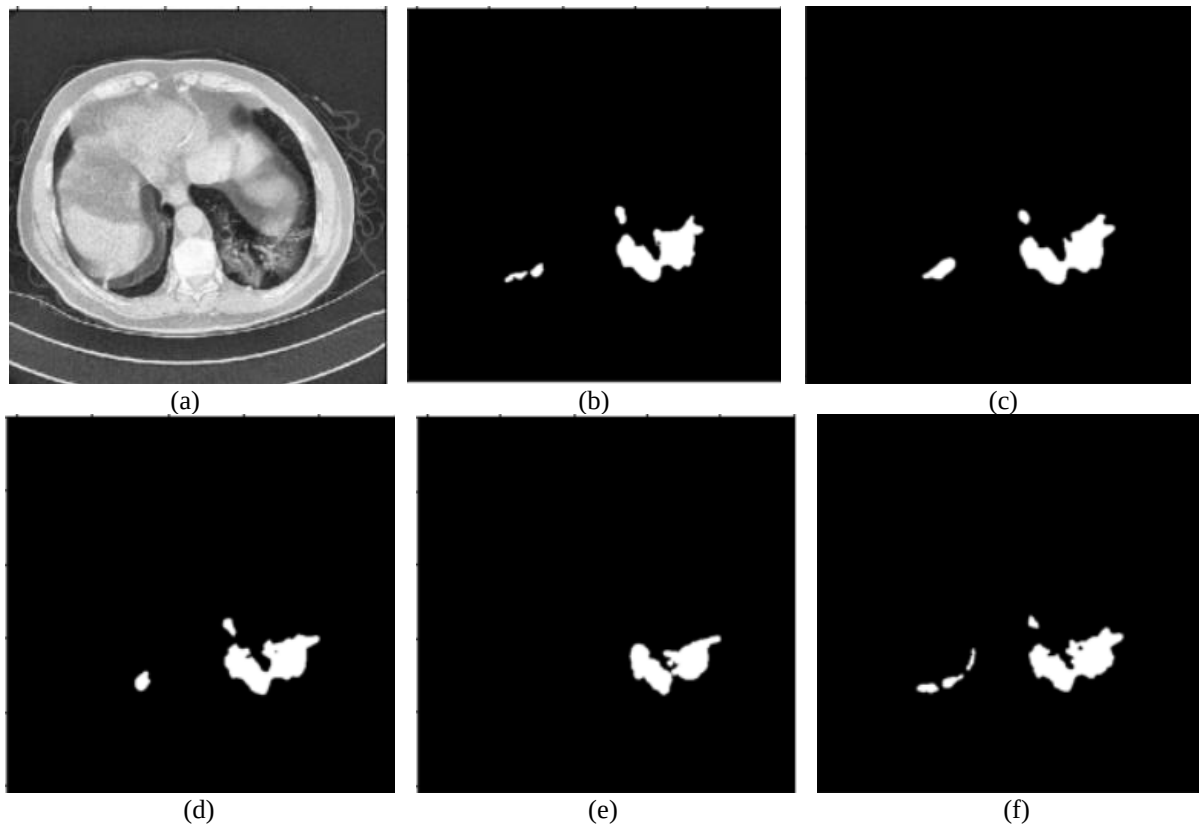


Figure. 11: (a) CT-scan data test second experiment, (b) ground truth data test. (c) is output model Resnet+Unet, (d) is output model Resnet+SwhsihUnet, (e) is output model EfficientNetB0+Unet, and (f) is output model EfficientNetB0 +SwishUnet,

Table 3. Performance the third experiment

Model	Accuracy (%)	Sensitivity (%)	Specificity (%)
Resnet+Unet [7]	**99.5**	77.8	**99.8**
Resnet+swishUnet	**99.5**	**79.8**	**99.8**
EfficientNetB0+Unet [10]	**99.5**	72.1	**99.8**
EfficientNetB0+swishUnet	99.3	70.1	**99.8**

5 Conclusion

In this paper, we propose SwishUnet to detect lesion COVID-19 by applying the swish activation in Unet, to get proper segmentation. We conducted an architecture evaluation using the 100 CT-scan and 9 volumes CT-scan from italian society of medical and interventional radiology. In this research, we do three experiments. The first experiment used only 54 CT scans that had lesion consolidation and GGO. The first experiment only used GGO and Consolidation lesions because Plural change lesions were only on 25 CT-scans. due to a lack of datasets with plural change causing the plural change to be undetectable. The second experiment used 99 CT-scan. In second experiment, we combined the three lesions into one lesion. the third experiment used a nine volumes CT-scan dataset, from this dataset we obtained 373 CT scans of COVID-19.

The first experiment results show the best sensitivity result is 82.7% using the Resnet+swishUnet method with a tversky loss function and a specificity is 98.9% using efficientNet+swishUnet with a dice loss function. The sensitivity and specificity of lesion consideration, Resnet+swishUnet with loss function weighted cross-entropy has the best sensitivity with 73.5% and Resnet+swishUnet softmax cross-entropy has the best specificity for lesion consideration with 99.1%.

In the second experiment, the best performance sensitivity is efficientUnet+Unet with 88.1% but it differs only a few percent when compared to the second best where the sensitivity of Efficient+swishUnet is 87.7%. The third experiment we get the best performance sensitivity is Resnet+swishUnet with 79.8%. All models have

same specificity where the value is 99.8%. from the experiment we have conclusion our proposed method with SwishUnet encoder has better performance than previous method.

Our future work of this study is to develop a multi-task deep learning for the classification and segmentation of Covid-19 infections on chest CT images. The proposed method can be improved by using the attention gate (AG) because it is still lacking in detecting small lesions and still incorrectly detects some noise as lesions.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Akhmad Irsyad made contributions to the experiments, deep learning architecture, the proposed approach, and the creation of the first draft. Handayani Tjandrasa provided supervision in conceptualizing and formulating frameworks, supervision in experimental and analytical process steps, preparation and editing of articles. Shintami Chusnul Hidayati provided supervision for the conceptualization, formulation of the problem, execution of the research process, and creation of the article.

References

[1] Z. Ye, Y. Zhang, Y. Wang, Z. Huang, and B. Song, "Chest CT manifestations of new coronavirus disease 2019 (COVID-19): a pictorial review" , *European Radiology*, Vol. 30, No. 8, pp. 4381–4389, 2020, doi: 10.1007/s00330-020-06801-0.

[2] A. Ghaffari, R. Meurant, and A. Ardakani, "COVID-19 Serological Tests: How Well Do They Actually Perform?", *Diagnostics*, Vol. 10, No. 7, Jul. 2020, doi: 10.3390/diagnostics10070453.

[3] A. Scohy, A. Anantharajah, M. Bodéus, B. K. Mukadi, A. Verroken, and H. R. Villalobos, "Low performance of rapid antigen detection test as frontline testing for COVID-19 diagnosis", *Journal of Clinical Virology*, Vol. 129, Aug. 2020, doi: 10.1016/j.jcv.2020.104455.

[4] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions", In: *Proc. of 6th International Conference on Learning Representations, ICLR 2018 - Workshop Track Proceedings*, 2018.

[5] H. Lu, Y. She, J. Tie, and S. Xu, "Half-UNet: A Simplified U-Net Architecture for Medical Image Segmentation", *Frontiers in Neuroinformatics*, Vol. 16, p. 54, 2022, doi: 10.3389/FNINF.2022.911679/BIBTEX.

[6] B. Baheti, S. Innani, S. Gajre, and S. Talbar, "Eff-UNet: A novel architecture for semantic segmentation in unstructured environment", In: *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, vol. 2020, pp. 1473–1481, 2020, doi: 10.1109/CVPRW50498.2020.00187.

[7] A. Abedalla, M. Abdullah, M. A. Ayyoub, and E. Benkhelifa, "Chest X-ray pneumothorax segmentation using U-Net with EfficientNet and ResNet architectures", *PeerJ Computer Science*, Vol. 7, pp. 1–36, 2021, doi: 10.7717/PEERJ-CS.607/SUPP-1.

[8] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", In: *Proc. of 32nd International Conference on Machine Learning, ICML 2015*, Vol. 1, pp. 448–456, Feb. 2015. Accessed: Dec. 12, 2020, [Online]. Available: https://arxiv.org/abs/1502.03167v3

[9] M. Uçar, "Automatic segmentation of COVID-19 from computed tomography images using modified U-Net model-based majority voting approach", *Neural Computing and Applications*, Vol. 34, No. 24, pp. 21927–21938, 2022, doi: 10.1007/s00521-022-07653-z.

[10] P. Bizopoulos, N. Vretos, and P. Daras, "Comprehensive Comparison of Deep Learning Models for Lung and COVID-19 Lesion Segmentation in CT scans", *arXiv*, 2020, doi: 10.48550/arxiv.2009.06412.

[11] L. Eckart, S. Eckart, and M. Enke, "A brief comparative study of the potentialities and limitations of machine-learning algorithms and statistical techniques", *E3S Web of Conferences*, Vol. 266, pp. 1–16, 2021, doi: 10.1051/e3sconf/202126602001.

[12] R. E. Putra, H. Tjandrasa, and N. Suciati, "Severity Classification of Non-Proliferative Diabetic Retinopathy Using Convolutional Support Vector Machine", *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 4, 2020, doi: 10.22266/ijies2020.0831.14.

[13] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation", *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 9351, pp. 234–241, 2015, doi: 10.1007/978-3-319-24574-4_28.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", In

Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2016-December, pp. 770–778, Dec. 2016, doi: 10.1109/CVPR.2016.90.

[15] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions", In: *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1–9, Oct. 2015, doi: 10.1109/CVPR.2015.7298594.

[16] M. Tan and Q. V Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks", *PMLR*, May 2019. Accessed: Jun. 08, 2021. [Online]. Available: http://proceedings.mlr.press/v97/tan19a.html

[17] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning", In: *Proc. of the Thirty-First AAAI Conference on Artificial Intelligence*, pp. 4278–4284, 2017.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", In: *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[19] R. H. R. Hahnioser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit", *Nature 2000 405:6789*, Vol. 405, No. 6789, pp. 947–951, 2000, doi: 10.1038/35016072.

[20] J. He, L. Li, J. Xu, and C. Zheng, "Relu deep neural networks and linear finite elements", *Journal of Computational Mathematics*, Vol. 38, No. 3, pp. 502–527, 2020, doi: 10.4208/JCM.1901-M2018-0160.

[21] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)", 2018, Accessed: May 15, 2022. [Online]. Available: http://arxiv.org/abs/1803.08375

[22] S. Jadon, "A survey of loss functions for semantic segmentation", In: *Proc. of 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology*, 2020, doi: 10.1109/CIBCB48159.2020.9277638.

[23] Dr. A. U. Ruby, P. Theerthagiri, D. I. J. Jacob, and Y. Vamsidhar, "Binary cross entropy with deep learning technique for Image classification", *International Journal of Advanced Trends in Computer Science and Engineering*, Vol. 9, No. 4, pp. 5393–5397, 2020, doi: 10.30534/ijatcse/2020/175942020.

[24] Q. Zhu, Z. He, T. Zhang and W. Cui, "Improving classification performance of softmax loss function based on scalable batch-normalization", *Applied Sciences (Switzerland)*, Vol. 10, No. 8, pp. 1–8, 2020, doi: 10.3390/APP10082950.

[25] S. Jadon, "A survey of loss functions for semantic segmentation", In: *Proc. of 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB 2020*, 2020, doi: 10.1109/CIBCB48159.2020.9277638.

[26] D. U. N. Qomariah, H. Tjandrasa, and C. Fatichah, "Segmentation of Microaneurysms for Early Detection of Diabetic Retinopathy Using MResUNet", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 3, 2021, doi: 10.22266/ijies2021.0630.30.

[27] I. Rizwan, I Haque, and J. Neubert, "Deep learning approaches to biomedical image segmentation", *Informatics in Medicine Unlocked*, Vol. 18, p. 100297, 2020. doi: 10.1016/j.imu.2020.100297.

[28] Italian Society of Medical and Interventional Radiology, "COVID-19 - Medical segmentation", 2020. http://medicalsegmentation.com/covid19/ (Accessed Nov. 05, 2021).

[29] N. Saeedizadeh, S. Minaee, R. Kafieh, S. Yazdani, and M. Sonka, "COVID TV-Unet: Segmenting COVID-19 chest CT images using connectivity imposed Unet", *Computer Methods and Programs in Biomedicine Update*, Vol. 1, p. 100007, 2021, doi: 10.1016/J.CMPBUP.2021.100007.

[30] J. Schlemper, … D. Rueckert, "Attention gated networks: Learning to leverage salient regions in medical images", *Medical Image Analysis*, Vol. 53, pp. 197–207, 2019, doi: 10.1016/J.MEDIA.2019.01.012.