Image segmentation of skin cancer using MobileNet as an encoder and linknet as a decoder

M Widiansyah, S Rasyid, P Wisnu, and A Wibowo
Informatics Departement, Science and Math Faculty, Diponegoro University, Jalan Prof. H.Soedarto, SH. Tembalang, Semarang 50275

Corresponding author: bowo.adl@live.undip.ac.id

Abstract. Cancer causes many deaths in the world. One of the most dangerous types of cancer is skin cancer caused by cancer cells that grow and spread uncontrollably in the skin layers. There are an estimated that 100,000 new cases of skin cancer will occur in the United States, which will cause around 6000 deaths by 2020. The survival rate of people with skin cancer can increase if successfully treated at an early stage. Examination of the lesion or dermoscopy images takes a long time and is prone to errors due to differences of opinion by doctors. A computer program for image processing was developed to improve accuracy and efficiency that helps doctors diagnose and evaluate. The most crucial stage in the diagnosis process is the segmentation of skin lesions. Wrong segmentation results will affect accuracy at the classification stage. This study developed a segmentation model for a skin cancer image that uses the MobileNet model as an encoder block and Linknet model for a decoder block by selecting the best hyper-parameter value from several training scenarios. MobileNet encoder and Linknet as a decoder are known to have the same ability, which is to overcome the problem of computational efficiency without compromising accuracy. Four hyper-parameters used are learning rate, the number of epochs, image size, and using pre-trained or not. Besides, there is a performance comparison scenario between the MobileNetV1 and MobileNetV2 encoder blocks. The model trained using the 2017 ISIC Challenge dataset consisting of 2000 training data, 150 validation data, and 600 test data. The Intersection over Union score obtained was 71.5% from models trained with a hyper-parameter learning rate of 0.001, 50 epochs, with an image size of 256 × 256 × 3, using a pre-trained model, and using a MobileNetV1 encoder block. The mask image that is the result of the segmentation of skin cancer images using the MobileNet encoder model and Linknet decoder is accurate enough to be used in the process at a later stage.

1. Introduction
Cancer causes many deaths in the world, one of the most dangerous is skin cancer. As the name implies, skin cancer is cancer caused by the development of cancer cells in the skin layer, which then grow and spread without control [1]. There is an estimation that in 2020 will be 100,350 new cases of melanoma in the United States and will cause the death of as many as 6,850 people in 2020 due to melanoma skin cancer [2]. Dermoscopy technique was developed to improve the performance of skin cancer diagnosis. However, manual examination of dermoscopy images usually takes a long time, is prone to errors and is highly subjective, where a trained dermatologist can produce a variety of diagnostic results [3]. A computer program was developed to detect melanoma from a dermoscopy image [4] automatically and solve the problem of misdiagnosis. Computer-Aided Diagnosis System is an important decision support
tool in medical imaging for diagnosis and evaluation [5]. Computer-Aided Diagnosis system to detect skin cancer has several steps, one of which is the image segmentation stage [6], [7].

In recent years, deep learning methods have developed very rapidly for image segmentation of skin cancer. In 2015 [8] proposed a fully convolutional network architecture consisting of an encoder block and a decoder block to perform segmentation tasks on biomedical images by eliminating pooling operations and replacing them with up-sampling operations. Furthermore, [9] in 2017 also proposed an encoder-decoder based architecture for the segmentation of skin lesions. Moreover, research conducted by [10] also suggests an architecture for image segmentation consisting of an encoder block resulting from a modified VGG16 model so that it is more efficient. The emergence of a fully convolutional network makes the decoder block a significant role in repairing the missing image features in the encoder block.

Apart from accuracy, efficient computing is also an essential aspect of segmentation. The effectiveness depends on the model design, including the width and depth of the network, which must involve many operations and parameters [11]. Models that have many parameters and long inference time are not suitable to be applied to mobile devices which require image processing at speeds of more than ten frames per second (fps) [12]. A study conducted by [13] proposed a framework for comparing multiple architectures for real-time segmentation. One of the comparisons made in this study is to compare the effect of several encoders, including MobileNet and ShuffleNet. The results obtained indicate that the MobileNet encoder has higher accuracy with more efficient computing. MobileNet is an architecture that is built with a depthwise separable convolution and has efficient computation so that it can be applied to mobile devices [14]. Many network models have been proposed to overcome the problem of effectiveness, one of which is the LinkNet model which explains that this model is ten times faster and has higher accuracy than the SegNet model [15]. The LinkNet model consists of an encoder block and a decoder block which have great accuracy without sacrificing processing time. The MobileNet encoder block will be combined with the Linknet decoder block to solve the problem of effectiveness and has good accuracy.

This paper will describe the development of an encoder-decoder model for skin cancer segmentation using the MobileNet encoder block and in the decoder section using Linknet. Then the model that is built will be trained and determine the best hyper-parameter such as learning rate, number of epochs, image size and pre-trained models.

2. Method

2.1. Dataset

The International Skin Imaging Collaboration (ISIC) is an international activity organized to improve melanoma diagnosis and reduce the risk of dying from skin cancer. The International Society sponsors ISIC for Digital Imaging of the Skin (ISDIS). ISIC-Archive contains a dataset of publicly available skin cancer images. As of 2019, the ISIC-Archive contains more than 23,700 dermoscopy images collected from internationally leading clinical centers obtained from various devices in each of these clinical centers.

Participation in extensive image collection with international coverage can ensure that these images represent clinically relevant relevance. All imagery that goes to ISIC-Archive is tested and inspected to ensure privacy and quality. Most of the images have clinical metadata that has been examined by melanoma experts. More than 13,000 images have also been annotated masks by skin cancer experts (ISIC isic-archive.com, 2019).

Training data and validation data are used for the training process, while test data are used for the testing process. The ISIC 2017 Skin Lesion Challenge dataset has 2750 data that has rules for dividing the dataset, namely 2000 training datasets, 150 validation datasets, and 600 remaining data are used for testing data. The stage after collecting the dataset is pre-processing the data. The dataset that has been obtained has various image sizes, so it is necessary to preprocess the dataset. Data pre-processing is performed by resizing the entire image to $128 \times 128 \times 3$ pixels and $256 \times 256 \times 3$ pixels.
2.2. Model
The model used for skin cancer segmentation is the MobileNet encoder model-Linknet decoder, which consists of two parts, namely MobileNet as the encoder block and Linknet as the decoder block. The complete architecture of MobileNetV1 and MobileNetV2 can be seen in figure 1 and figure 2.

![Figure 1. MobileNetV1 Architecture](image1.png)

![Figure 2. MobileNetV2 Architecture](image2.png)

Training data and validation data are used for the training process, while test data are used for the testing process. The ISIC 2017 Skin Lesion Challenge dataset has 2750 data that has rules for dividing the dataset, namely 2000 training datasets, 150 validation datasets, and 600 remaining data are used for testing data. The stage after collecting the dataset is pre-processing the data. The dataset has been obtained has various image sizes, so it is necessary to pre-process the dataset. Data pre-processing is performed by resizing the entire image to 128 × 128 × 3 pixels and 256 × 256 × 3 pixels.

2.3. MobileNet-Linknet model
The International Skin Imaging Collaboration (ISIC) is an international activity organized to improve melanoma diagnosis and reduce the risk of dying from skin cancer. The International Society sponsors ISIC for Digital Imaging of the Skin (ISDIS). ISIC-Archive contains a dataset of publicly available skin cancer images. As of 2019, the ISIC-Archive contains more than 23,700 dermoscopy images collected from internationally leading clinical centers obtained from various devices in each of these clinical centers.

The layer used in the Linknet decoder has several blocks, with each block having a convolution, up-sampling, and two convolution operations then. The last block layer is followed by a 3 × 3 convolution operation and the Sigmoid activation function. The complete Linknet decoder architecture can be seen in Figure 3. The development of the MobileNet encoder model-Linknet decoder is done using a library, namely Keras. Keras is used to solve problems related to neural networks (Keras, 2020). Keras provides...
models and block layers for neural networks. Examples of layer blocks used in this study include the input layer, zero-padding, convolution, batch normalization, ReLU6, depthwise convolution, pointwise convolution, concatenation up-sampling, and sigmoid activation.

![Figure 3. Linknet Architecture](image)

2.4. Evaluation metric
Intersection over Union is one of the evaluation metrics used for object detection or segmentation. Intersection over Union is measured using the similarity and diversity of a data set. In general, Intersection over Union is calculated using Equation 1 according to research conducted by [16] with $T$ is ground-truth and $P$ is prediction value. F1-score, commonly known as dice score, is one of the evaluation metrics for object detection or segmentation and intersection over union [17]. In general, the f1-score or dice score can be calculated using Equation 2 with $T$ is ground-truth and $P$ is prediction value.

$$IoU = \frac{\text{Intersection}}{\text{Union}} = \frac{|T \cap P|}{|T \cup P|}$$ (1)

$$F1 - score = \frac{2 \times \text{Intersection}}{\text{Union} + \text{Intersection}} = \frac{2|T \cap P|}{|T| + |P|}$$ (2)

3. Result and discussion
We conducted experiments by dividing into scenarios stages, which are described in detail in this subsection.

3.1. Scenario 1
The first scenario is done to determine the hyper-parameter value that produces the highest score in the MobileNetV1 encoder model - the Linknet decoder. The hyper-parameter value in question is the value of the learning rate, the number of epochs, the image size and whether or not it is using a pre-trained model. Hyper-parameter values are obtained from the combination of defined hyper-parameters. The learning rates used in the first scenario are 0.001 and 0.0001. This learning rate determination refers to the research conducted by [18] which. Then the epochs value refers to a study conducted by [19] who used the epochs value of 50 epochs. So the epochs value for this first scenario is 50 epochs. To determine the effect of differences in epochs value and whether there is a better epoch value, two new epochs values will be added, namely 250 epochs and 500 epochs. Then for the image size in this scenario using image sizes that do not require too long computation, namely pixels and pixels. Then the pre-trained model aims to determine the effect of the use of pre-trained imagenet with those not using pre-trained.

The experimental results of the first scenario produced 24 scores for each given combination of hyper-parameters. The model's ability is evaluated using the IoU score and f1-score. Table 1 shows the results of the model testing for the combination of each hyper-parameter. Based on the results of the test...
scores in Table 3, it shows that the highest test score for the first scenario is a combination of hyper-parameter number seven. The hyper-parameter learning rate of 0.001, the number of epochs 250, the image size of $256 \times 256 \times 3$, and the pre-trained image yields the best model with the highest score IoU score of 69.84% and the highest f1-score of 81.31%.

### Table 1. Result of Scenario 1 based on test data

| No | lr   | epochs | Image-size | IoU   | f1-score |
|----|------|--------|------------|-------|----------|
| 1  | 0.001| 50     | $128 \times 128 \times 3$ | 69.54% | 81.16%   |
| 2  | 0.001| 50     | $256 \times 256 \times 3$ | 67.61% | 79.66%   |
| 3  | 0.001| 250    | $128 \times 128 \times 3$ | 65.87% | 78.51%   |
| 4  | 0.001| 250    | $256 \times 256 \times 3$ | 69.84% | 81.31%   |
| 5  | 0.001| 250    | $128 \times 128 \times 3$ | 64.01% | 76.88%   |
| 6  | 0.0001| 50     | $256 \times 256 \times 3$ | 65.87% | 78.51%   |
| 7  | 0.0001| 250    | $128 \times 128 \times 3$ | 66.46% | 78.93%   |
| 8  | 0.0001| 250    | $256 \times 256 \times 3$ | 68.46% | 80.29%   |
| 9  | 0.0001| 50     | $128 \times 128 \times 3$ | 62.03% | 75.15%   |
| 10 | 0.0001| 50     | $256 \times 256 \times 3$ | 69.55% | 81.11%   |
| 11 | 0.0001| 250    | $128 \times 128 \times 3$ | 65.84% | 78.44%   |
| 12 | 0.0001| 250    | $256 \times 256 \times 3$ | 69.37% | 81.06%   |
| 13 | 0.0001| 50     | $128 \times 128 \times 3$ | 56.48% | 70.52%   |
| 14 | 0.0001| 50     | $256 \times 256 \times 3$ | 60.03% | 73.62%   |
| 15 | 0.0001| 250    | $128 \times 128 \times 3$ | 69.08% | 80.84%   |
| 16 | 0.0001| 250    | $256 \times 256 \times 3$ | 59.15% | 72.95%   |
| 17 | 0.0001| 50     | $128 \times 128 \times 3$ | 69.04% | 80.65%   |
| 18 | 0.0001| 50     | $256 \times 256 \times 3$ | 58.34% | 72.09%   |
| 19 | 0.0001| 250    | $128 \times 128 \times 3$ | 68.89% | 80.66%   |
| 20 | 0.0001| 250    | $256 \times 256 \times 3$ | 59.86% | 73.71%   |
| 21 | 0.0001| 50     | $128 \times 128 \times 3$ | 69.83% | 81.26%   |
| 22 | 0.0001| 50     | $256 \times 256 \times 3$ | 58.55% | 71.99%   |

3.1.1. Comparison of different learning rate values. The score values of the 24 experimental hyper-parameter value combinations in Table 1 will be grouped based on the same learning rate values of 0.001 and 0.0001. Table 2 shows the standard deviation value and the mean f1-score for each learning rate. Based on the average score and standard deviation in Table 2, it can be seen that the learning rate of 0.001 produces an average f1-score that is higher than the learning rate of 0.0001, namely 79.19% and results in a smaller standard deviation value of 0.0183.

### Table 2. Average and standard deviation IoU score

| Learning rate | IoU           | f1 – score       |
|---------------|---------------|------------------|
|               | Average       | Std. Dev.        | Average | Std. Dev. |
| 0.001         | 0.6701        | 0.02372          | 0.79188 | 0.01827   |
| 0.0001        | 0.64025       | 0.05361          | 0.76713 | 0.04307   |

3.1.2. Comparison of different epoch values. Testing the effect of the number of epochs is carried out by grouping the test score based on the number of epochs, then a training chart is displayed of the model that has the best test score for each epoch. Figures 4 show the graph of the model training that has the best score of each epochs value.
Figure 4 shows models trained up to 500 epoch validation score scores tend to remain and show no increase in validation scores. The validation scoring using epoch 500 also shows the same results as epoch 50 and epoch 250 which ranges from 81% - 86%. Based on learning charts with different epoch values it can be known that the validation score of models trained with epoch 50 is not much different with epoch 250 or epoch 500. This indicates that a high epoch value does not necessarily make the validation score of the model. Even after the epoch value touches the number 200 the movement of the validation score chart is not as good as at the time of the first 200 epoch. It can therefore be concluded that epochs 50 is the best value because the epochs value of 50 has been able to produce the best validation score and does not take long in the training process.

3.1.3. Comparison of different input size. The Table 3 presents the average score scoring as well as the standard deviation for each image size. The standard deviation value and average of the image size deviation score of 256 $\times$ 256 $\times$ 3 is smaller than the standard value of the image size deviation score of 128 $\times$ 128 $\times$ 3. The smaller the standard value of the deviation score obtained in this test means that the model is more stable against the hyper-parameter of the size of the image. Because image sizes 256 $\times$ 256 $\times$ 3 and 128 $\times$ 128 $\times$ 3 do not show significant differences, then both of these values will be retested in the second scenario.

Table 3. Average and standard deviation IoU score

| Input Size   | IoU $f_1$ – score |
|--------------|-------------------|
|              | Average | Std. Dev. | Average | Std. Dev. |
| 128 $\times$ 128 | 0.65148 | 0.04521 | 0.7767 | 0.03598 |
| 256 $\times$ 256 | 0.65887 | 0.04256 | 0.78231 | 0.03442 |

3.1.4. Comparison of different testing type. The training score and validation that using the pre-trained model have a higher score than those who do not use the pre-trained model. The difference occurred very significantly during the first 100 epochs where the score of the model using the pre-trained model increased rapidly and was approaching a stable score with only 50 epochs of epochs. A very drastic difference can also be seen at the time of the first epochs where the score for model training using the pre-trained model has reached a score of 85% while the validation score has also touched 80%. The model without using the pre-trained model where the training score at the first epochs is only 64% and also the validation score is only able to reach 50%. Table 4 shows the most stable model is a model that is trained using the pre-trained model which has a smaller standard deviation and higher average score than the model that does not use the pre-trained model. This low standard deviation score means that the model is more stable. Therefore, training and testing for the next scenario will use a pre-trained model.
Table 4. Average and Standard Deviation IoU Score

| Pre-trained | IoU Average | IoU Std. Dev. | f1 - score Average | f1 - score Std. Dev. |
|-------------|-------------|---------------|--------------------|---------------------|
| Yes         | 0.69185     | 0.00614       | 0.8083             | 0.00458             |
| No          | 0.6185      | 0.03398       | 0.7507             | 0.02854             |

3.2. Scenario 2

The learning rate value in the second scenario uses the same value as in the first scenario. Then the experiment was added by giving new learning rate values, namely 0.1 and 0.01 and resulted in four different learning rate values. So that for the second scenario in the first stage, an experiment will be carried out with four different learning rate values and with the best image size in the first scenario, namely 256 × 256 × 3 pixels. Then in the second stage the experiment will be carried out using the best learning rate value in the first stage of the experiment. Furthermore, an experiment was conducted to retest the effect of image size on accuracy scores. The image size tested is the same size as in the first scenario, namely 128 × 128 × 3 and 256 × 256 × 3 pixels and uses a pre-trained model, the epoch hyper-parameter value, and the best learning rate in the previous stage and scenario. The score in the first scenario is obtained from the model obtained from the last epochs of training. For the second scenario, in the last stage, a comparison will be made between the score of the model at the time of the last epochs with the value of the model score when the model gets the highest validation score. Overall, for each learning rate value and the same image size at each stage, training experiments and testing will be carried out seven times.

3.2.1. Comparison of different learning rate values. Based on the average test score in Figure 5, it can be seen that the highest average score was obtained when the model was trained using a learning rate of 0.001, namely 81.5% for the f1-score and 70.15% for the IoU score, while for the learning rate 0.1, the average score is the lowest. Figure 5 also shows the standard deviation value of the mean score for each learning rate. The smallest standard deviation score indicates that the model is the most stable. Based on the score and standard deviation obtained, the value of 0.001 will be used as the learning rate value for the next stage and scenario.

3.2.2. Comparison of different input size. The highest average score for both f1-score and IoU is a model trained using an image size of 256 × 256 × 3, which has an average f1-score of 81.5% and has an average IoU score of 70.15%. Table 5 shows the mean score and standard deviation of each image size. Based on the average score and standard deviation in Table 7, the model with the smallest standard deviation score is a model trained using an image size of 256 × 256 × 3. The most stable model is a

![Figure 5. IoU scores and standard deviation each learning rates](image-url)

(b)
model trained using an image size of $256 \times 256 \times 3$ which results in the smallest standard deviation of scores. Therefore, the best image size in this second stage will be used for training at the next stage and scenario.

### Table 5. Average and standard deviation IoU score

| Input Size | IoU   | $f_1$ - score |
|------------|-------|---------------|
|            | Average | Std. Dev. | Average | Std. Dev. |
| $256 \times 256$ | 0.7015 | 0.006634641 | 0.815302857 | 0.005121645 |
| $128 \times 128$ | 0.695334286 | 0.017468058 | 0.809657143 | 0.012649935 |

3.2.3. **Comparison of different testing type.** Up to this stage, the test carried out is a model test when it reaches the last epochs. Therefore, the results of the best hyper-parameters in the previous stage are retested on the model when the model reaches the highest validation score. Table 6 shows that the highest average score is owned by the model when it reaches the highest validation score. The stability of the resulting accuracy score also shows that the model score with the best validation has a stable score, as evidenced by the small standard deviation value compared to the standard deviation value of the model score at the time of the last validation.

### Table 6. Average score and standard deviation

| Testing Mode       | IoU   | $f_1$ - score |
|-------------------|-------|---------------|
|                   | Average | Std. deviation | Average | Std. Deviation |
| Last Validation    | 0.7015 | 0.006634641 | 0.815302857 | 0.005121645 |
| Best Validation    | 0.7068314 | 0.0064856 | 0.8198943 | 0.0048828 |

3.3. **Scenario 3**

This third scenario will conduct training and testing to determine the effect of the MobileNetV1 and MobileNetV2 encoders on the test score. Therefore, the third scenario will carry out a training and testing process using different types of encoders, namely MobileNetV1 and MobileNetV2 and using the best hyper-parameter values in the previous scenario.

### Table 7. Testing Score on MobileNetV1, dan MobileNetV2

| No | MobileNetV1 | MobileNetV2 | $f1$-score | $f1$-score |
|----|-------------|-------------|------------|------------|
| 1  | 0.711       | 0.823       | 0.684      | 0.801      |
| 2  | 0.696       | 0.812       | 0.700      | 0.813      |
| 3  | 0.702       | 0.817       | 0.698      | 0.813      |
| 4  | 0.709       | 0.821       | 0.705      | 0.818      |
| 5  | 0.715       | 0.826       | 0.713      | 0.823      |
| 6  | 0.701       | 0.815       | 0.699      | 0.813      |
| 7  | 0.714       | 0.825       | 0.711      | 0.822      |

The results of the seven times test and training scores in Table 7 show that the majority of the MobileNetV1 encoders produce higher test scores than the MobileNetV2 encoders. The MobileNetV1 encoder produced the highest IoU score of 71.5% while the MobileNetV2 encoder produced the highest IoU score of 71.3%. The mean score and standard deviation of the test in Table 8 shows that the mean score of the MobileNetV1 and MobileNetV2 encoders does not have such a big difference, namely 0.81676 and 0.81602. It's just that for the stability of the score even though the difference is very small, MobileNetV1 is more stable than MobileNetV2.
Table 8. Average score and standard deviation

| Test Mode     | IoU | $f_1$ score |
|---------------|-----|-------------|
|               | Average | Std. Deviation | Average | Std. Deviation |
| MobileNetV1   | 0.70683 | 0.00649     | 0.81989 | 0.00488       |
| MobileNetV2   | 0.70145 | 0.00887     | 0.81474 | 0.00686       |

3.4. Evaluation stage

Evaluation is a stage to determine the best hyper-parameter value seen from the highest score results when testing the model. The model made aims to perform the task of segmenting skin cancer images trained using the ISIC 2017 Skin Lesion Challenge dataset. The model that has the highest accuracy is the Linknet model which is trained using the MobileNetV1 encoder with a hyper-parameter learning rate of 0.001 with an epochs of 50 and an image size of 256×256 and uses the pre-trained imagenet model.

![Figure 6. Plotting the Noisy Image (top), low Contrast (middle), bad ground-truth](image)

The mask prediction results generated in Figure 8 show that the model is still wrong to predict when the input image has noise in the area around the lesion in the form of hair. Therefore, to overcome this problem it should be necessary to carry out a pre-processing process on the dataset to minimize the noise of hair around the skin lesion. The skin cancer image that has low contrast in the area of the skin cancer and healthy skin. Although the visible image of the skin cancer lesion looks the same color as healthy skin and looks like it is not part of skin cancer, the ground-truth image shows that the area that resembles healthy skin is part of the skin cancer lesion. Therefore, we need a method at the pre-processing stage of data to increase the color contrast between the skin cancer lesion area and the healthy skin area so that it is expected to increase the model's ability to segment. The images of skin cancer appear to have a contrasting and almost uniform color for healthy skin color and skin cancer lesion color respectively. However, the ground-truth image in the image did not cover all skin cancer lesions and only covered the edges of the skin lesions even though they appeared to have the same color.
4. Conclusion
The conclusion from the research that has been done is that the best performance of skin segmentation using the MobileNet encoder and LinkNet decoder models is obtained by training the model using hyper-parameter values, namely the learning rate value of 0.001 with the number of epochs of 50 epochs, then using the input image size \(256 \times 256 \times \) Model. The resulting version is built using the MobileNetV1 encoder which is the first version of MobileNet with transfer learning using a trained imagenet model. The mask image from the skin cancer image segmentation results using the MobileNet encoder and LinkNet decoder models is sufficient to be used in the next stage. There are several problems regarding the dataset in the segmentation process, the first is the noise in the image of skin cancer in the form of hair around the skin cancer lesion. The second problem is that there are some images that have a low contrast level between skin cancer lesions and healthy skin. The next problem is related to the ground-truth image that does not match the image of the skin cancer lesion, so it is necessary to re-validate the ground-truth image. Suggestions for further research are to build a model that is able to distinguish skin cancer images that have a low contrast level between skin cancer lesions and healthy skin and are able to deal with noise problems in the dataset, which can be overcome by increasing the number of datasets. Then it is necessary to re-validate the ground-truth image by an expert doctor. In addition, it is expected to be able to develop a model with a multi-scale input image size so that it can produce better accuracy. Further suggestions are expected for further research to implement this segmentation model into mobile devices.

Acknowledgements
This work was supported by the RISTEKDIKTI, The Republic of Indonesia for PENELITIAN DASAR Research Grant 2020 under Grant No. 257-22/UN7.6.1/PP/2020.

References
[1] Guerra-Rosas E and Álvarez-Borrego J 2015 Biomed. Opt. Expr. 6 10 3876
[2] Society A C 2020 Atlanta Am. Cancer Soc.
[3] Binder M et al. 1995 Arch. Dermatol. 131 3 286–291
[4] Javed R, Rahim M S M, Saba T and Rehman A 2020 Netw. Model. Anal. Heal. Informatics Bioinforma. 9 1
[5] Masood A and Al-Jumaily A A 2018 Int. J. Biomed. Imaging
[6] Goyal P K, Nirvikar and Jain M K 2018 Lect. Notes Networks Syst. 38 63–73
[7] Yu L, Chen H, Dou Q, Qin J and Heng P A 2017 IEEE Trans. Med. Imaging 36 4 994–1004
[8] Ronneberger O, Fischer P and Brox T 2015 Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 9351 234–241
[9] Ramachandram D and DeVries T 2017 LesionSeg: Semantic segmentation of skin lesions using Deep Convolutional Neural Network arXiv
[10] Badrinarayanan V, Kendall A and Cipolla R 2017 IEEE Trans. Pattern Anal. Mach. Intell. 39 12 2481–2495
[11] Zhao H, Qi X, Shen X, Shi J and Jia J 2018 Eur. Conf. Comput. Vis. 418–434
[12] Paszke A, Chaurasia A, Kim S and Culurciello E 2016 ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation pp. 1–10
[13] Siam M, Gamal M, Abdel-Razek M, Yogamani S, Jagarsand M and Zhang H 2018 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work. 2018-June pp. 700–710
[14] Howard A G et al. 2017 MobileNets: Efficient convolutional neural networks for mobile vision applications arXiv
[15] Chaurasia A and Culurciello E 2017 IEEE Vis. Commun. Image Process. VCIP 2017 2018-Jan pp. 1–4
[16] Rahman M A and Wang Y 2016 ISVC 1
[17] Bertels J et al. 2019 International Conference on Medical Image Computing and Computer-Assisted Intervention pp. 92–100
[18] Goyal M et al. 2019 IEEE Access 8 pp. 4171–4181
[19] Tang P et al. 2019 Comput. Methods Programs Biomed. 178 pp. 289–301