Indonesian Journal of Applied Statistics

E-ISSN 2621–086X

Vol. 5, No. 1, pp. 39-47, May 2022
https://jurnal.uns.ac.id/ijas/article/view/59442
https://doi.org/10.13057/ijas.v5i1.59442

Implementation of Transfer Learning for Covid-19 and Pneumonia Disease Detection Through Chest X-Rays Based on Web

Nindya Eka Apsari[*], Sugiyanto, Sri Sulistijowati Handajani
Statistics Study Program, Universitas Sebelas Maret, Surakarta, Indonesia

***Corresponding author:** nindyaekaapsari@student.uns.ac.id

Abstract. Coronavirus disease 2019, known as COVID-19, attacks the human respiratory system caused by severe acute respiratory syndrome coronavirus-2 (SARS-Cov-2). COVID-19 disease and pneumonia show similar symptoms such as fever, cough, and even headache. Diagnosis of pneumonia can be tested through diagnostic tests, including blood tests, chest X-rays, and pulse oximetry, while the diagnosis of COVID-19 recommended by WHO is with a swab test (RT-PCR). But in fact, the swab test method takes a relatively long time, about one to seven days, for the result, and is not cheap. For that, there needs to be a development that can be one of the options in diagnosing COVID-19 and pneumonia at once, especially since both diseases have similar symptoms. One option that can be done is the diagnosis using a chest X-ray. This research aims to detect COVID-19 disease and pneumonia through chest X-rays using transfer learning to increase the accuracy of disease diagnosis in a more efficient time. The architecture used is EfficientNet B0 with variations in optimization parameters, learning rates, and epochs. EfficientNet B0 Adam optimization with a learning rate of 0.001 in the 6th epochs is a great model that we obtained. Furthermore, the evaluation of the model got accuracy, precision, recall, and an f1-score of 92%. Then the model visualization is done using Grad-CAM. To implement the best model, web application development is done to make it easier to detect COVID-19 disease and pneumonia.

Keywords: COVID-19; EfficientNet; pneumonia; transfer learning; web

1. INTRODUCTION

Coronavirus disease 2019, better known as COVID-19, attacks the human respiratory system caused by the Severe Acute Respiratory Syndrome Coronavirus-2 (SARS-Cov-2). Common symptoms experienced by sufferers include fever, cough, anosmia, and muscle aches. Pneumonia is an infectious lung disease caused by bacteria, viruses, or fungi [1]. Symptoms caused by bacterial pneumonia include high fever, increased breathing and pulse, and bluish discoloration of the nails. Viral pneumonia has symptoms such as fever, flu, muscle aches, dry cough, headache, and weakness.

COVID-19 and pneumonia show similar symptoms. In diagnosing COVID-19, WHO recommends a swab test (RT-PCR), while a diagnosis of pneumonia can be carried out by diagnostic tests, including blood tests, chest X-rays, and pulse oximetry. But in reality, the swab test takes a relatively long time, which is one to seven days, and is not cheap. For this reason, it is necessary to have other alternatives that can be an option in the diagnosis of COVID-19 and pneumonia at the same time, especially since the two diseases have similar symptoms. One option that can be used is to use a chest X-ray image.

Submitted: 16 February 2022; ***Accepted:*** 31 May 2022

Several previous kinds of research have detected COVID-19 and pneumonia through chest X-rays using the Convolutional Neural Network (CNN) method. Zein [2] conducted research on detecting the COVID-19 virus in chest X-ray images using artificial intelligence-based deep learning on Python and obtained 92% accuracy. Another study by Estomihi [3] created a pneumonia detection application system through chest X-ray images and obtained an accuracy rate of 93.3%. In this research, the pre-processing stage uses grayscaling and CLAHE (Contrast Limited Adaptive Histogram Equalization) algorithm backpropagation as a classification method by conducting experiments at a maximum of 1000 epochs.

Convolutional Neural Network (CNN) is a method of developing an Artificial Neural Network (ANN), which is widely applied in image processing. CNN has had significant developments in object recognition and image classification compared to simpler methods [4]. Transfer learning is supervised machine learning for studying pre-trained models with fewer datasets [5]. In its application, one approach that can be chosen is fine-tuning. Fine-tuning will modify the parameters of the pre-trained model and train it for new tasks. Based on the background above, this research was conducted on the detection of COVID-19 and pneumonia through chest X-ray images using transfer learning to increase the accuracy of disease diagnosis in a more efficient time.

2. LITERATUR REVIEW

Many studies related to the detection of COVID-19 and pneumonia using X-ray images have been carried out previously. Research by Estomihi [3] created a pneumonia detection system through chest X-ray images to obtain an accuracy rate of 93.3%. In this study, the pre-processing used grayscaling and CLAHE (Contrast Limited Adaptive Histogram Equalization) algorithm backpropagation as a classification method by experimenting with a maximum of epochs 1000.

Singh [6] conducted a study on COVID-19 detection using the process of transfer learning model Support Vector Machine ensemble. This study uses CT-scan divided into two classifications, namely COVID and non-COVID. Training data, validation data, and test data are 432, 62, and 208 images. The accuracy obtained is 95.67%. Ardiawan [7] built an application to detect COVID-19 and pneumonia through chest X-ray images using the residual Convolutional Neural Network. This method applies transfer learning architecture Resnet-18 with epochs obtained at 94% accuracy.

2.1. Convolutional Neural Network

Convolutional Neural Network (CNN) is one of the development methods of ANN, which is widely applied to image processing. CNN has had significant developments in object recognition and image classification compared to simpler methods [4]. Artificial neural network feed-forward that is connected to each other and moves forward. The Backpropagation method is added to evaluate the weight updates carried out. Neurons in CNN consist of bias, weight, and activation function. The CNN architecture differentiates it from neural networks. It extracts important features through the convolution method, followed by a classification process. The steps of CNN can be shown in Figure 1.

Figure 1 shows that in general, there are two processing steps in CNN, namely feature learning (hidden layer) and classification. The first step of feature learning is the step of extracting image information to study the values contained in it. CNN performs classification using multiple extractions in one input. The number of extractions is stored in depth. The feature learning depends on the depth of an image. This process can be repeated until the desired matrix size is obtained. The values obtained from feature learning are then converted into vectors and entered in the classification step. At the classification step, the neural network will be used to classify objects based on their class.

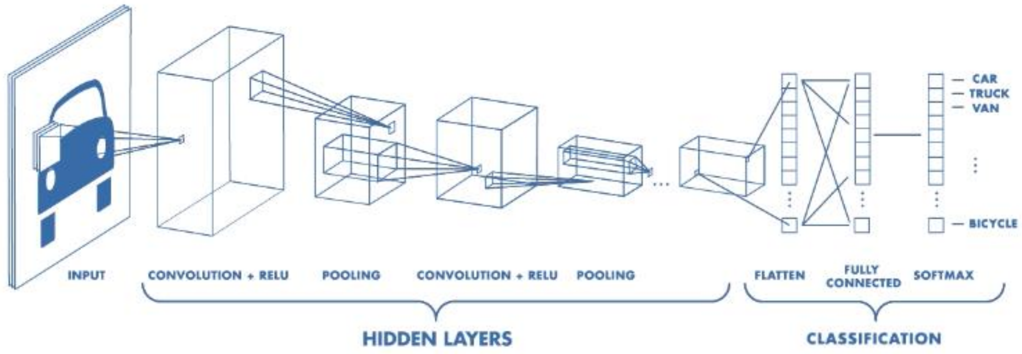


Figure 1. Convolutional Neural Network

2.2. EffientNet

EfficientNet is an architecture in the Convolutional Neural Network that can predict and classify images accurately [8]. EfficientNet architecture is composed of several blocks called Mobile-Net Convolution (MBConv). Table 1 shows that the EfficientNet has nine steps with certain blocks. Matrix filter kernel 3×3 has 32 channels with one layer with a resolution of 224×224 pixels. The second step consists of an MBConv block with an expansion factor value of 1 with a kernel filter measuring 3×3, which has 16 channels with one layer with a resolution of 112×112 pixels. These steps are repeated according to Table 1 until the 9th step, namely the convolution layer with a 1×1, pooling layer, and fully connected layer. The steps in the MBConv block can be shown in Figure 2.

Figure 2 indicates that the MBConv block is composed of multiple layers. The first layer is a convolution layer with a filter kernel measuring 1×1 and the number of filters is the product of the expansion factor and the number of channels from the input. ReLU6 is the activation function used in this layer. This layer is known as the expansion layer because it makes the feature map larger than the input. Then, the information is passed on to the depthwise convolution kernel filter 3×3 and the ReLU6 activation function and then forwarded to kernel filter 1×1 with a linear activation function.

Table 1. Architecture EfficientNet

Step	Operator	Resolution	Channels	Layer
1	Conv3x3	224×224	32	1
2	MBConv1, k3x3	112×112	16	1
3	MBConv6, k3x3	112×112	24	2
4	MBConv6, k5x5	56×56	40	2
5	MBConv6, k3x3	28×28	80	3
6	MBConv6, k5x5	14×14	112	3
7	MBConv6, k5x5	14×14	192	4
8	MBConv6, k3x3	7×7	320	1
9	Conv1x1 & Pooling & FC	7×7	1280	1

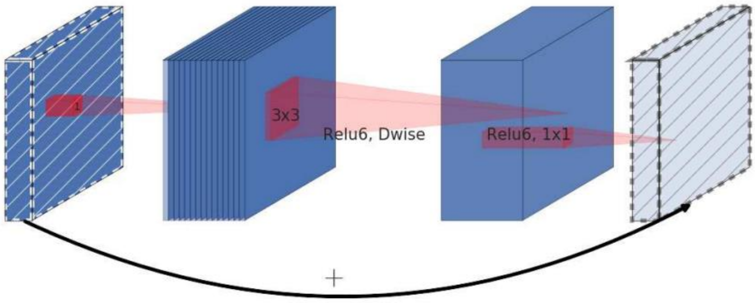


Figure 2. MBConv block

2.3. Transfer Learning

Transfer learning is supervised machine learning for studying pre-trained models with fewer datasets. In its application, there are various approaches that can be chosen. One of them is fine-tuning. Fine-tuning will modify the parameters of the pre-trained model and train it for new tasks. Fine-tuning can be an option when the data form is very different from the data in the previous architectural training. Transfer learning can be done by changing the last layer of the model by adjusting to the new data classification, initiating weight updates at that layer, then retraining with the learning rate. This retraining process is carried out on all parameters to minimize the loss of the new task [9].

2.4. Gradient-Weighted Class Activation Mapping (Grad-CAM)

Gradient-weighted Class Activation Mapping (Grad-CAM) is a method for visualizing the final activation in deep learning. Grad-CAM uses the gradient information on the last convolutional layer of the CNN to assign important values [10]. Grad-CAM is important to know how the model identifies classes based on the image used. Class weights are derived in this method using the following equation.

$$w_k^c = \frac{1}{Z}\sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}$$

feature Z indicates the number of parts of the map. This method uses a feature map generated by the gradient value of the classification, namely y^c, with respect to the feature map A_{ij}^k in a randomly selected layer. Making a heatmap refers to the following formula.

$$L_{Grad-CAM}^c = ReLU\left(\sum_k w_k^c A^k\right)$$

ReLU is applied to the linear combination heatmap because Grad-CAM only processes features that positively affect the class of interest, for example, pixels whose intensity must be increased to increase y^c. Negative pixels tend to fall into another category in images.

3. RESULTS AND DISCUSSION

This research used secondary data obtained from the Italian Society of Medical and Interventional Radiology (SIRM), Radiological Society of North America (RSNA), and Kaggle. The data has the number of x-ray images for each category, namely 3616 COVID-19 categories, 10,192 normal categories, and 1345 pneumonia categories. To avoid imbalanced data that may cause errors in interpreting the evaluation model, 1300 data were taken same for each category with a total of 3900 data images. The image size dimensions are 299 x 299 pixels with a .png file format. Figure 3 is the sample chest X-ray image used for training data. Figure 3 shows samples from each category. There are three normal images, two pneumonia images, and 1 COVID-19 image. If observed, each of these image categories looks similar to one another. For this reason, modeling is carried out to be able to classify existing images.

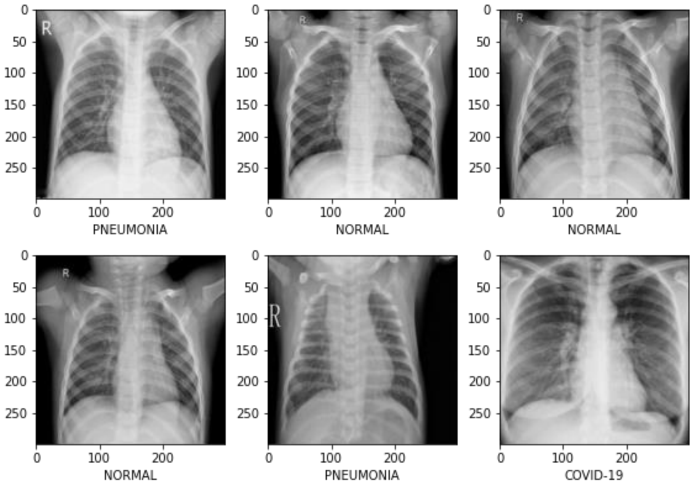


Figure 3. Chest X-rays sample images

Data pre-processing is divided into three parts, namely training, validation, and test data. Training data is data used to train the model, while validation data is data used for the validation process to prevent overfitting. Test data is used to test the model as an implementation of the use of the model in real cases. Test data must never be used before by the model. The distribution of this data follows a ratio of 80%:10%:10%, namely training data of 1040 images, validation data, and test data of 130 images each.

The pre-processing is to perform data augmentation. Data augmentation is a technique of manipulating data without losing the essence or essence of the data. This is done to make the model more resilient to slight variations and therefore prevent the model from being overfitted. Table 2 contains details of the augmentation data carried out. The zoom range is to enlarge the image with a value of 0.25. Horizontal flip is the process of turning the image from the left side to the right side and vice versa. The shear range is to tilt the image with a slope value of 0,2. Based on the details of the augmentation data carried out, Figure 4 is the result of the augmentation data carried out.

Table 2. Augmentation data

Augmentation	Description
Zoom range	0.25
Horizontal flip	Yes
Shear range	0.20

Figure 4 shows the results of the augmented image. Augmentation makes one image multiplied as much as the augmentation process is carried out. Next, resize the image, which was originally 299 x 299 pixels, to 224 x 224 pixels. The process is done to adjust the needs of the input model used. Data image is obtained clean, meaning that the data is ready for modeling.

The modeling carried out in this research uses the EfficientNet B0. EfficientNet B0 is an architecture that has been previously trained and tested with other data. The implementation of model training was carried out on existing chest X-ray image data from this architecture. The process of training this model in this research is called transfer learning with EfficientNet B0 as the pre-trained model. The use of transfer learning aims to obtain effective and efficient data training. This study uses a fine tuning that modifies the parameters of the pre-trained model and trains it to work on the new training.

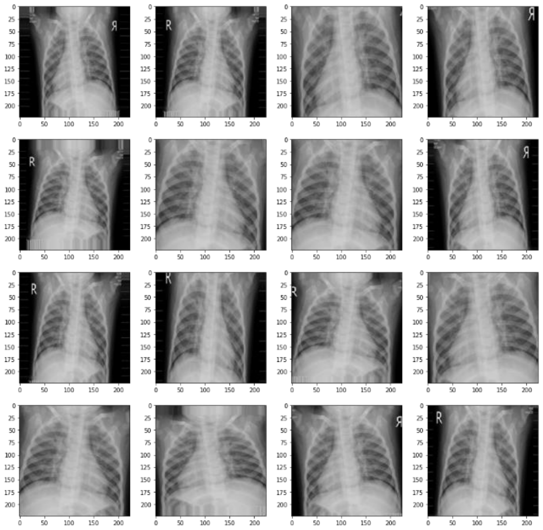


Figure 4. Augmentation data

Model training uses three parameters to obtain the best model, namely optimization, learning rate, and epochs. There are two optimizations tried, namely Adam and Stochastic Gradient Descent (SGD). Learning rates are 0.01 and 0.001. Epochs from each model are 15 epochs with a total of 98 steps per epoch. The selection of the best model is seen from the value of accuracy and loss of validation data. The results are shown in Table 3.

Table 3. Model training

Optimization	Learning Rate	Epochs	Val Accuracy	Val Loss
		(step per epoch = 98)		
Adam	0.010	4	97.50%	1.0421
	0.001	6	97.81%	0.0986
SGD	0.010	8	98.12%	0.4347

Table 3 shows that the optimization of SGD with a learning rate of 0.01 in lowest the 8th epoch has the highest validation accuracy of 98.12%, while Adam's optimization owns the loss with a learning rate of 0.001 in epoch the 6th. To see a clearer comparison among models, they are presented in Figures 5.

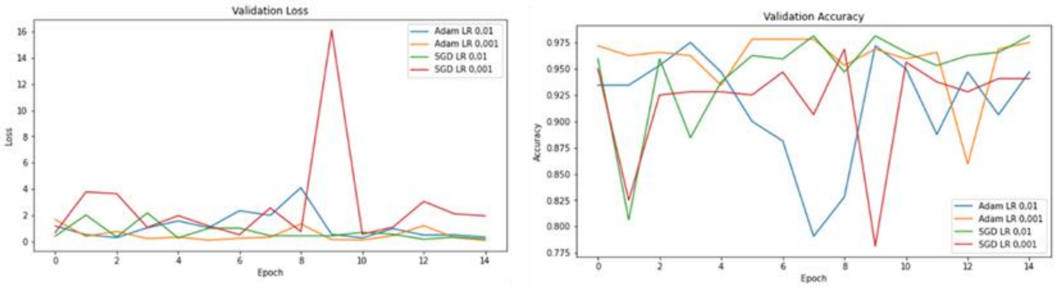


Figure 5. Evaluation model

Figure 5 shows the comparison of the validation loss and accuracy of the models. Validation loss has a stable value is Adam's optimization with a learning rate of 0.001. The higher the validation loss can be an indication of an overfitting model. For this reason, it is better to choose a model with the loss a minimum and stable. The model that has a stable validation accuracy value is the Adam optimization model with a learning rate of 0.001. Based on the loss and validation accuracy that has been obtained, the best model is chosen, namely using Adam's optimization with a learning rate of 0.001. This model is then implemented into a web to detect the presence of COVID-19 or pneumonia through a chest X-ray.

The best model that has been obtained in the previous process then evaluates the model using test data which has a number of each class of 130 images of chest X-ray data. The model evaluation results obtained that the model has an accuracy of 92%. This value can be good for further implementation in detecting COVID-19 and pneumonia. In addition, the average value of precision, recall, and f1-score is 92%. Precision means that the proportion of correctly predicted labels from the total predictions in all classes has an average of 92%. Recall, namely, the ratio of labels that are classified as positive from the total class that is positive in the model has an average of 92%. In comparison, the f1-score is a combination of precision and recall.

Gradient-weighted Class Activation Mapping, commonly known as Grad-CAM is a method for visualizing activation in the CNN model. Grad-CAM uses the gradient information on the last convolutional layer of the CNN to assign important values [10]. Grad-CAM is important to know how the model identifies classes based on the image used.

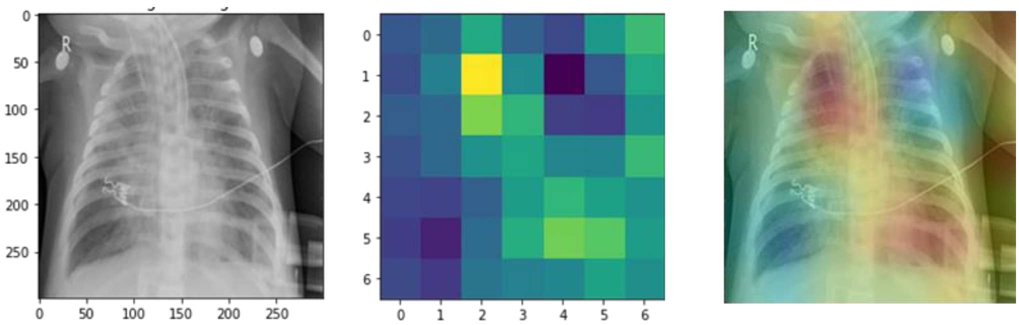
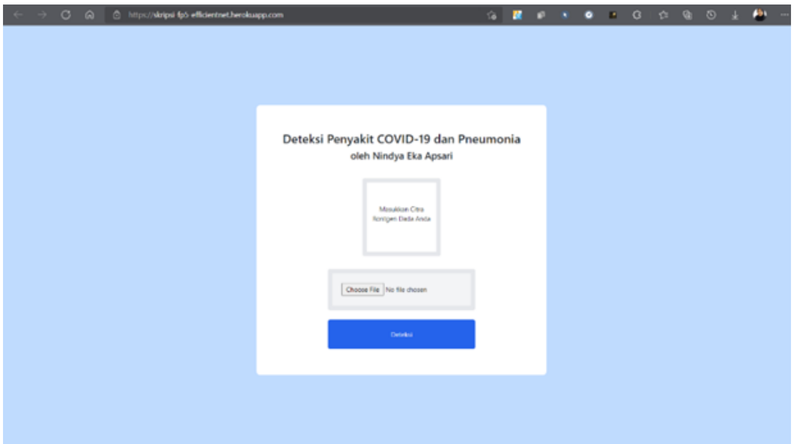


Figure 6. Grad-CAM

Figure 6 shows how the model sees the image and classifies it. The left image is the initial image of the chest X-ray image. The middle image is a heatmap that shows the amount of activation in the last layer of the model. The right image shows the merging of the visualizations between the two previous images. This is to make it easier to interpret the heatmap that has been obtained. The color gradation shows how much activation the model has on that pixel. The brighter the color gradation, the stronger the model activation on the pixel. Color gradations can also be used to reference doctors to detect parts of the lungs affected by COVID-19 or pneumonia.

Implementation of the model into the web application makes it easier to detect COVID-19 or pneumonia through chest X-ray images. The best model obtained is EfficientNet B0 with Adam optimization parameters and a learning rate of 0.001. The model is implemented on the web using the libraries Flask and Heroku (Figure 7). The web application can be accessed via the link https://skripsi-fp5-efficientnet.herokuapp.com/.



https://skripsi-fp5-efficientnet.herokuapp.com

Deteksi Penyakit COVID-19 dan Pneumonia
oleh Nindya Eka Apsari

Masukkan Citra Rontgen Dada Anda

Figure 7. Web application (https://skripsi-fp5-efficientnet.herokuapp.com/)

4. CONCLUSIONS

EfficientNet B0 model with Adam's optimization and learning rate of 0.001 is very good in detecting COVID-19 and pneumonia. Evaluation of the model gives accuracy, precision, recall, and an f1-score of 92%. Furthermore, the model is implemented into a web application to make it easier to detect COVID-19 and pneumonia through chest X-ray images. The web application is built using Flask and Heroku and can be accessed via the link https://skripsi-fp5-efficientnet.herokuapp.com/.

REFERENCES

[1] A. E. Thompson, "Pneumonia is a lung infection that can be caused by bacteria, viruses, or fungi," Jama Patient Page Pulmonary, vol. 315, no. 6, p. 2016, 2016.

[2] A. Zein, "Pendeteksian Virus Corona Dalam Gambar X-Ray Menggunakan Algoritma Artifical Intelligence Dengan Deep Learning Python," J. Teknol. Inf. ESIT, vol. XV, no. 01, pp. 19–23, 2020.

[3] J. Estomihi, Klasifikasi Penyakit Pneumonia dari Citra X-Ray Menggunakan Backpropagation, Repos. Institusi USU, 2019.

[4] L. Tobias, A. Ducournau, F. Rousseau, G. Mercier, and R. Fablet, "Convolutional Neural Networks for object recognition on mobile devices: A case study," Proc. - Int. Conf. Pattern Recognit., vol. 0, pp. 3530–3535, 2016, doi: 10.1109/ICPR.2016.7900181.

[5] H. W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler, "Deep learning for emotion recognition on small datasets using transfer learning," ICMI 2015 - Proc. 2015 ACM Int. Conf. Multimodal Interact., pp. 443–449, 2015, doi: 10.1145/2818346.2830593.

[6] M. Singh, S. Bansal, S. Ahuja, R. K. Dubey, B. K. Panigrahi, and N. Dey, "Transfer learning–based ensemble support vector machine model for automated COVID-19

detection using lung computerized tomography scan data," Med. Biol. Eng. Comput., vol. 59, no. 4, pp. 825–839, 2021, doi: 10.1007/s11517-020-02299-2.

[7] R. Ardiawan, T. Rakasiwi, G. R. Maulana, and N. Yudistira, "Aplikasi Deteksi Covid-19 dan Pneumonia melalui Citra X-ray Dada menggunakan Residual Convolutional Neural Network," Researchgate, no. January, 2021, doi: 10.13140/RG.2.2.16582.91202.

[8] L. T. Duong, P. T. Nguyen, C. Di Sipio, and D. Di Ruscio, "Automated fruit recognition using EfficientNet and MixNet," Comput. Electron. Agric., vol. 171, February, p. 105326, 2020, doi: 10.1016/j.compag.2020.105326.

[9] Z. Li and D. Hoiem, "Learning without Forgetting," IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 12, pp. 2935–2947, 2018, doi: 10.1109/TPAMI.2017.2773081.

[10] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization," Rev. do Hosp. das Cl??nicas, vol. 17, pp. 331–336, 2016, [Online]. Available: http://arxiv.org/abs/1610.02391.