 ISSN: 2579-7298

A Mobile Deep Learning Model on COVID-19 CT-Scan Classification

Prastyo Eko Susanto [1], Arrie Kurniawardhan [2], Dhomas Hatta Fudholi [3], Ridho Rahmadi [4]

[1]*Master Program in Informatics, Department of Informatics, Universitas Islam Indonesia, Yogyakarta, Indonesia*
[2,3,4]*Department of Informatics, Universitas Islam Indonesia, Yogyakarta, Indonesia*

[1] *prastyo.susanto@students.uii.ac.id* ; [2] *arrie.kurniawardhani@uii.ac.id* ; [3] *hatta.fudholi@uii.ac.id* ;
[4] *ridho.rahmadi@uii.ac.id*

ARTICLE INFO

Article history:
Received 02 Nov 2021
Revised 26 Jan 2022
Accepted 11 Mar 2022

Keywords:
ct scan;
covid-19;
deep learning;
MobileNet;
classification

ABSTRACT

COVID-19 pandemic is currently happening in the world. Previous studies have been done to diagnose COVID-19 by identifying CT-scan images through the development of the novel Joint Classification and Segmentation System models that work in real-time. In this study, the author focuses on a different motivation and innovation focused on the development of mobile deep learning. MobileNet, a deep learning model as a method for classifying the disease COVID-19, is used as the base model. It has a good level of efficiency and reliability to be implemented on devices that have small memory and CPU specifications, such as mobile phones. The used data in this study is a CT-scan image of the lungs with a horizontal slice that has been classified as positive or negative for COVID-19. To give a broader analysis, the author compares and evaluates the model against other architectures, such as MobileNetV3 Large, MobileNetV3 Small, MobilenetV2, ResNet101, and EfficientNetB0. In terms of the developed mobile architecture model, the classification of COVID-19 using MobileNetV2 obtained the best result with 0.81 accuracy.

I. Introduction

According to the WHO, there have been 199,466,211 confirmed cases of COVID-19 with 4,244,541 deaths as of August 4, 2021 [1]. The COVID-19 becomes a pandemic because it has a worldwide reach and affects a large geographic area. The coronavirus was first announced in Indonesia on March 2020. As of August 26, 2020, there are 160,165 coronavirus positive people, of whom 115,409 have recovered and the remaining 6,944 have died [2].

Wu et al. did research on COVID-19 classification using CT Scan images. As illustrated in Figure 1, they developed a revolutionary Joint Classification and Segmentation System (JCS) that works in real-time and may be utilized to diagnose COVID-19. Certain explanatory classification or segmentation models cannot be fully applied as comprehensive characteristics for COVID-19 diagnosis. To do so, the segmentation model adds to the study by recognizing entire lung lesions and determining the severity of COVID-19 patients. However, experienced radiologists are not allowed to make substantial segmentation label annotations. They created a COVID-19 diagnostic system based on shared categorization and explanatory segmentation models to better integrate its benefits for better applications [3].

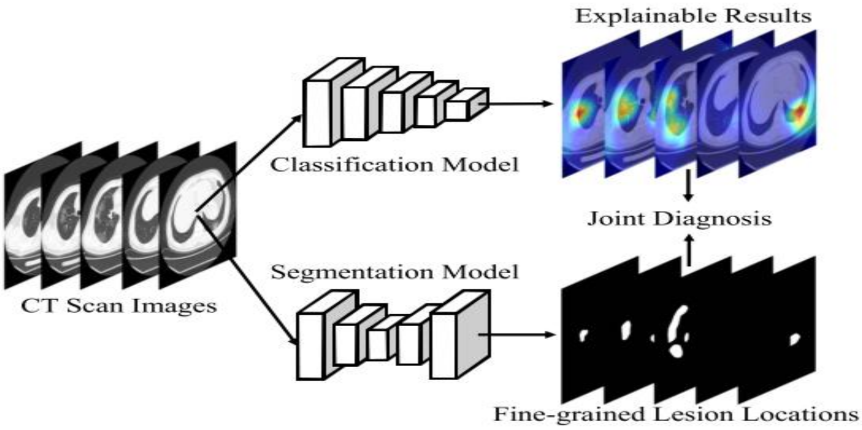

Fig. 1. Illustration of research carried out by [3]

Deep learning is a type of machine learning that makes use of Artificial Neural Networks (ANNs). There are three kinds of learning: supervised, semi-supervised, and unsupervised. We wish to try something new in this research. We developed a classification model for COVID-19 CT scan images using MobileNet, a mobile base architecture. The architecture is a state-of-the-art picture feature extractor with a high level of efficiency, making it suitable for implementation on devices with limited memory and CPU, such as mobile phones.

The research will vary from prior research in that it will focus on developing a deep learning classification model with a tiny memory footprint that can be used on smartphones. Because of the smaller but more reliable model [4], the focus of this study will be on mobile-based development. We developed a COVID-19 illness categorization based on CT scan pictures to assess whether individuals are positive or negative for COVID-19. For the training and testing process, we chose certain photos. The following models were compared: MobileNetV2, MobileNetV3 Large, MobileNetV3 Small, EfficientNetB0, and ResNet101. Following that, we chose the finest models for smartphones, particularly Android, from a pool of five models.

II. Literature Review

Deep learning is a field of machine learning that is inspired by the human brain and attempts to model neural networks using abstract modeling using a series of non-transformation functions linearly stacked in layers [5]. In order for models to comprehend the categorization pattern of each training set, deep learning in the classification process requires training data [6].

During the COVID-19 pandemic, it was incredibly difficult to diagnose COVID-19 patients using CT-Scan data since CT-Scan data is highly confidential. The authors based their findings on data from [7], which has been confirmed by radiologists who have discovered and treated COVID-19 patients since the outbreak began. The collection includes 349 photographs of COVID-19 positive patients and 397 photos of COVID-19 negative patients. On this dataset, studies that used multi-task and self-supervised learning algorithms achieved an F1-score of 0.90 and an AUC of 0.98 [7].

In a prior study, He et al. compared the performances of three models in categorizing COVID-19, one of which is DenseNet-169, and the other is ResNet-50 [8]. They propose the Self-Trans technique, which combines contrastive self-supervised learning with transfer learning to build strong and unbiased feature representations while minimizing the danger of overfitting. The F1-score of the research technique is 0.85, and the AUC is 0.94. The goal of the paper is to reduce the number of inefficiencies in the process of testing COVID-19 patients, as well as to relieve the burden on medical professionals, by developing a deep learning model that can automatically interpret and read CT scan images in classifying whether the patient is COVID-19 positive or not [8].

Cohen et al. conducted research on COVID-19, the data of which was appropriated by CT scans of the lungs [9]. Data will be utilized to build and test a deep learning-based model in this research. When compared to other kinds of pneumonia, COVID-19 characteristics can help predict survival. Other COVID-19 research uses an artificial intelligence technique, and all approaches employ binary categorization. ResNet, ResNet50, InceptionV3, and VGG19 are some of the artificial intelligence-based algorithms for detecting COVID-19 that have been proposed in the literature. In this study [10], the VGG19 architecture received a score of 98 percent, the ResNet architecture received a score of 96 percent, the ResNet50 architecture received a score of 95 percent, and the InceptionV3 architecture received a score of 96 percent.

III. Data And Methodology

Data collection, modeling, and evaluation are the three main steps recommended in the mobile deep learning model study on CT-Scan COVID-19 lung classification. As seen in Figure 2, the detailed process after collecting the data is comprises of the following steps: splits the dataset, train the models, evaluates each model, compare all models' performance, and choose the best model.

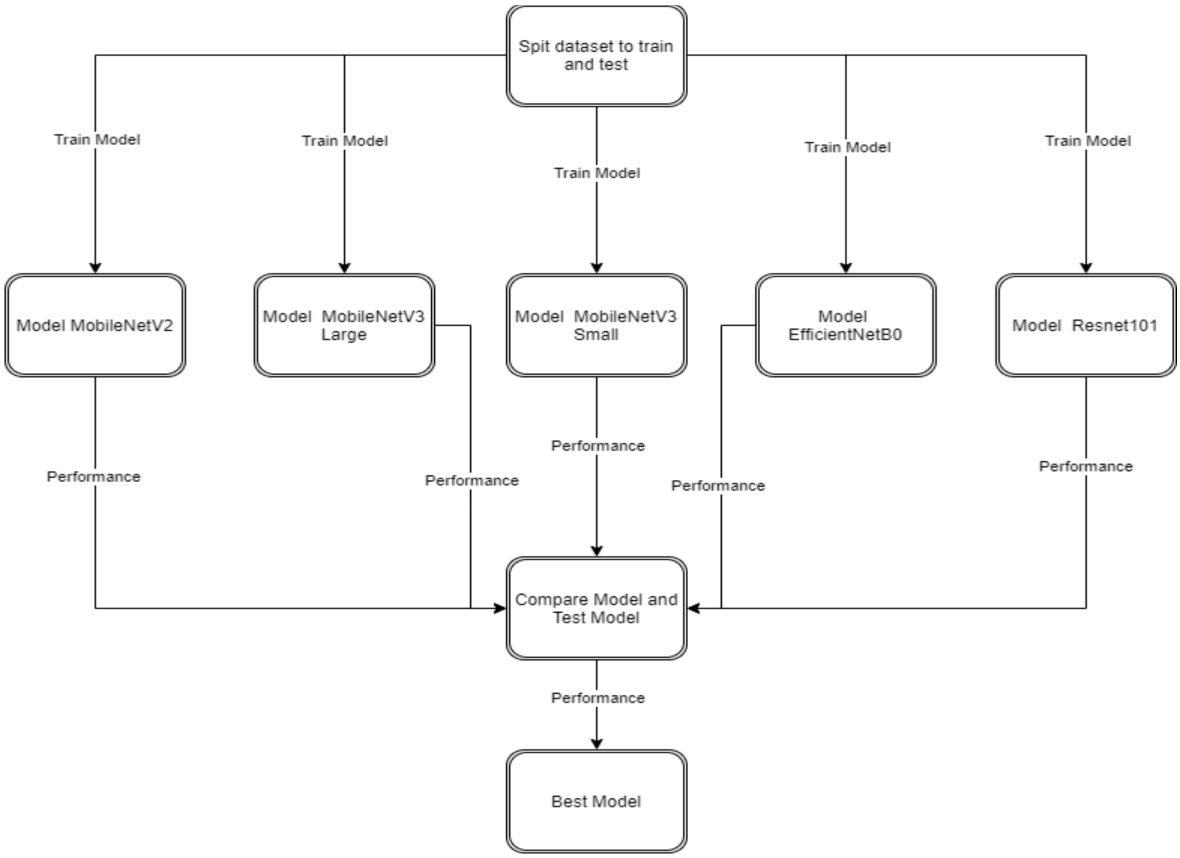

Fig. 2. Research methodology.

A. Data Collection

Data is collected in the form of horizontal CT-scan pictures of the lungs to identify patients as positive or negative for COVID-19. The information gathered comes from a variety of sources, including the Italian Medical Radiology and Intervention (SIRM) [11]. SIRM also published COVID-19 axial and coronal CT files on Radiopedia [12]. The European Society of Radiology (ESR) [13] is another source for CT-scanned pictures of COVID-19 patients.

Yang et al. created a COVID-CT dataset from this separate data, which included 349 COVID-19 CT-Scan pictures from 216 patients and 397 non-patient CT-scans for COVID-19 [7]. This data

ISSN: 2579-7298

collection, which has been gathered and made accessible at https://github.com/UCSD-AI4H/COVID-CT, belongs to prominent radiologists who have identified and treated COVID-19 patients since the outbreak of the pandemic. Figure 3 shows an example of positive COVID-19 CT-scan picture, whereas Figure 4 shows an example of negative COVID-19 CT-scan image.

Fig. 3. Example of Positive COVID-19 CT Scan Results [7]

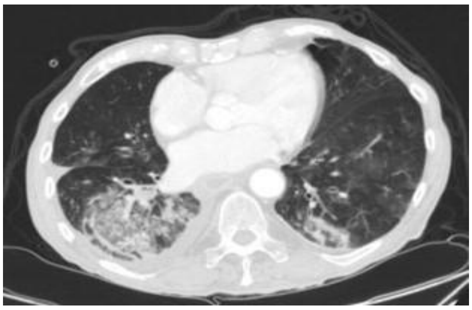

Fig. 4. Example of COVID-19 Negative CT Scan Results [7]

B. Modeling

Modeling is the technique of learning or training CT-Scan pictures in order to classify whether or not someone is a Covid sufferer. This procedure is performed on an NVIDIA GeForce GTX 1050 using 191 positive COVID-19 training data and 234 negative COVID-19 training data. To test or choose a model, there are test data for 98 positive and 105 negative COVID-19 patient data.Figure 5 depicts the procedure.

We firstly set the hyperparameter before the training process begin. The batch size used in this experiment are tuned between 16, 32, or 64. The input image size is set to be 160 x 160 pixel. During the experiment, the data is going through an augmentation process that rotates the data image by about 40 degrees. Rotation is the most used setup in data augmentation strategy. The source picture is rotated by several degrees clockwise or counterclockwise at random, causing the object's position in the frame to vary [14]. Data augmentation is the act of changing or editing images in such a way that the computer recognises the updated image as a new image but people can still identify it is the same image [15]. There are five base architectures that we used in our experiment. They are MobileNetV3 Large, MobileNetV3 Small, MobilenetV2, ResNet101, and EfficientNetB0, as seen in Figure 2.

MobileNetV3 was created using a mix of neural architecture search (NAS) algorithms and manual layer adjustments from MobileNetV2. MobileNetV3 is set up for implementation on low-spec devices like mobile phones through a network architecture search process, also known as a neural architecture search (NAS), equipped with the algorithm NetAdapt, and then enhanced through the benefits of a new component, namely the nonlinear computation process h-swish, a modified version of previous non-linear swish computations on MobileNetV2. MobileNetV3 features two models: MobileNetv3-Large and MobileNetv3-Small, which are designed for heavy and low resource use cases, respectively [16].

In image categorization, ResNet is a convolutional neural network that closely resembles human vision. From end to end, the network of extraction functions begins with low, medium, and high-level classifiers in multi-layer mode, and the number of stacked layers can enrich the function's "level." ResNet stands for Residual Network, which is a deep residual network implementation. Classification mapping becomes more optimum with residual learning reformulation, as the weight of various nonlinear layers is attempted to zero through residual values to approach goal classification identity mapping [17].

EfficientNet, which stands for Efficient Network, is a development of the Architecture Deep Learning Model MNasNet, but it has been improved in terms of size model optimization and accuracy. When compared to other ConvNets with the same accuracy, EfficientNet models require less parameters and FLOPS (Floating Points Performance Per Second). The model's scaling is carefully investigated to precisely balance the network's depth, width, and resolution, allowing for improved performance. All depth/width/resolution dimensions are scaled uniformly using efficient coefficients. EfficientNet is obtained via Neural Architecture Search (NAS) [18].

C. Model Evaluation

The researcher then used CT-Scan experimental data to evaluate a COVID-19 classification model. The accuracy, accuracy, recall, and F1 scores of the MobileNetV3 Large, MobileNetV3 Small, MobilenetV2, ResNet101, and EfficientNetB0 classification models were measured as part of the evaluation. The procedure of determining the evaluative value was carried out on five models as a parameter and a measure of model. The success factor of the evaluation is when the compared predicted label and actual data labels is match, either positive COVID-19 or negative COVID-19. The evaluation step tells researchers as to whether the application satisfies expectations.

During evaluation, there are two things, which we may need to point out and to give more careful attention. They are false positive result and false negative. False positive is the result that gives positive result, but they are not actually sick. In the other hand, false negatives refer to those who has negative result when in fact they are sick. The patient may experience undue worry because of the false positive, leading to a costly and intrusive diagnostic or even unneeded therapy (overtreatment). If the test is a false negative, on the other hand, the patient will feel protected, even if he has a condition. As a result, the diagnosis is delayed, and the necessary treatment is not carried out, resulting in increased morbidity and death [19].

The metrics that we used in this research as the evaluation are precision, recall, and F1-score. Precision (1) refers to the model's precision or accuracy in prediction [20]. By classifying them as such classes, recall (2) determines how many real classes the model captures. Because there are only two labels in this study, Positive COVID-19 or Negative COVID-19, the F1 (3) score will be a reasonable metric to employ if Precision and Recall need to be balanced.

$$\text{Precision} = \frac{\text{True Class}}{\text{True Class + False Predicted Class}} = \frac{\text{True Class}}{\text{Total Predicted Class}} \quad (1)$$

$$\text{Recall} = \frac{\text{True Class}}{\text{True Class + False Predicted Other Class}} = \frac{\text{True Class}}{\text{Total Actual Class}} \quad (2)$$

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

IV. Result and Discussion

The dataset used for the experiment has a total of 425 images as the training data and 203 images for the testing data. The training data consist of 191 CT-scan images of positive COVID-19 and 234

CT-scan images of negative COVID-19. In the other hand, testing data consist of 98 CT-scan images of positive COVID-19 and 105 CT-scan images of negative COVID-19.

Five classifier architectures, namely MobileNetV3 Large, MobileNetV3 Small, MobilenetV2, ResNet101, and EfficientNetB0, were trained using 425 images to generate a model. Then, each model was validated using 203 images and measured using Precision, Recall, F1-score, and Accuracy, as shown in Table 1. To find the best configuration, the batch size was varied to 16, 32, and 64 for each classifier. The 0 and 1 codes, written after the metric, are used to indicate negative and positive for Covid-19, respectively.

Table 1. Evaluation metrics

Architecture	***Precision -0***	***Precision-1***	***Recall-0***	***Recall-1***	***F1-score-0***	***F1-score-1***	***Acc***	***Batch***
MobileNetV3 Large	0.78	0.80	0.83	0.74	0.80	0.77	0.79	size 16
	0.75	0.77	0.80	0.71	0.77	0.74	0.76	size 32
	0.70	0.81	0.87	0.60	0.77	0.69	0.74	size 64
MobileNetV3 Small	0.68	0.80	0.87	0.56	0.76	0.66	0.72	size 16
	0.64	0.81	0.90	0.47	0.75	0.59	0.69	size 32
	0.74	0.82	0.86	0.68	0.80	0.74	0.77	size 64
MobileNetV2	0.75	0.83	0.87	0.69	0.81	0.76	0.78	size 16
	0.81	0.82	0.84	0.79	0.82	0.80	0.81	size 32
	0.77	0.88	0.91	0.70	0.83	0.78	0.81	size 64
EfficientNetB0	0.81	0.73	0.71	0.82	0.76	0.77	0.76	size 16
	0.75	0.77	0.80	0.71	0.77	0.74	0.76	size 32
	0.73	0.75	0.78	0.69	0.76	0.72	0.74	size 64
ResNet101	0.75	0.80	0.84	0.69	0.79	0.74	0.77	size 16
	0.73	0.77	0.81	0.65	0.77	0.72	0.75	size 32
	0.69	0.90	0.94	0.55	0.80	0.68	0.75	size 64

Precision, recall, and F1-score were calculated using (1), (2), and (3) respectively. Those evaluation metrics were calculated according to data provided in the confusion matrix [21], as shown in Table 2. In the confusion matrix, the data is investigated whether the data is True Positive (TP), True Negative (TN), False Positive (FP), or False Negative (FN). True Positive (TP) data is the number of positive data that is detected as positive data, while True Negative (TN) data is the number of negative data that is detected as negative data. False Positive (FP) data is the number of negative data that is detected as positive data, and False Negative (FN) data is the number of positive data that is detected as negative data [22]. CP is total data that predicted correctly, otherwise I is total data that predicted incorrectly.

Precision is defined as the proportion of True Positive (TP) and the data that is predicted as positive data [23]. In contrast, the definition of Recall is a comparison between True Positive (TP) and the data that is actually positive [23]. The F1-Score is a harmonic mean of Precision and Recall [23]. The best F1-Score is 1.0, while the worst F1-Score is 0 [23]. In terms of representation, if F1-Score has a good score, it indicates that our model has good precision and recall [23].

Table 2. Comparison Results

Architecture	*TP*	*FP*	*TN*	*FN*	*CP*	*I*	*Batch*
MobileNetV3 Large	73	18	87	25	160	43	size 16
	70	21	84	28	154	49	size 32
	59	14	91	39	150	53	size 64
MobileNetV3 Small	55	14	91	43	146	57	size 16
	46	11	94	52	140	63	size 32
	67	15	90	31	157	46	size 64
MobileNetV2	68	14	91	30	159	44	size 16
	77	17	88	21	165	38	size 32
	69	9	96	29	165	38	size 64
EfficientNetB0	80	30	75	18	155	48	size 16
	70	21	84	28	154	49	size 32
	68	23	82	30	150	53	size 64
ResNet101	68	17	88	30	156	47	size 16
	67	20	85	31	152	51	size 32
	54	6	99	44	153	50	size 64

According to Table 1, MobileNetV2 can achieve the highest average Precision, average Recall, average F1-score, and accuracy by 0.83, 0.82, 0.81, and 0.81, respectively. For Precision-1, the highest number that can be achieved in this experiment is 0.90 by ResNet101. It means the positive rate that is detected by ResNet101 is more precise than the other models. While the highest Recall-1 can be achieved by EfficientNetB0, 0.82.

Table 2 shows the detailed number of data detected correctly and incorrectly. MobileNetV2 can predict the data more accurately than the other models, 165 out of 203 test images. MobileNetV2 with batch size 32 identifies a CT-scan image faster than it with batch size 64. The difference is approximately 10 second.

V. Conclusion

The researchers classified patients as positive or negative for COVID-19 using CT-scan data from the lungs with horizontal slices. The goal of the research is to give an overview of the possibility in classifying COVID-19 based on lungs CT-scan through mobile deep learning architecture. In this research, we use five base architectures to be experimented, MobileNetV3 Large, MobileNetV3 Small, MobilenetV2, ResNet101, and EfficientNetB0. We will assess and test the performance of each deep learning model. The COVID-19 classification result, specifically MobileNetV2 with a batch size of 32, has a precision of 0.81. It is the best result compared to another model. During the testing, it achieved 165 correct predictions and 21 false negatives. The greatest correct prediction rating and the lowest false-negative rating produce the best model outcomes. These results were acquired by comparing models from multiple current architectures that have not yet been implemented on smartphones. As our future works, we will implement the model on mobile devices.

References

[1] W. H. Organization, “WHO Coronavirus (COVID-19) Dashboard,” *World Health Organization*, 2021. https://covid19.who.int/ (accessed Aug. 04, 2021).

[2] S. T. P. COVID-19, “Peta Sebaran,” *26 Agustus 2020*, 2020. https://covid19.go.id/peta-sebaran (accessed Aug. 26, 2020).

[3] Y.-H. Wu *et al.*, “JCS: An Explainable COVID-19 Diagnosis System by Joint Classification and Segmentation,” pp. 1–11, 2020, [Online]. Available: http://arxiv.org/abs/2004.07054.

[4] Y. Wang *et al.*, “A survey on deploying mobile deep learning applications : A systemic and technical perspective,” *Digit. Commun. Networks*, no. June 2020, 2021, doi: 10.1016/j.dcan.2021.06.001.

[5] A. Santoso and G. Ariyanto, “Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah,” *Emit. J. Tek. Elektro*, vol. 18, no. 01, pp. 15–21, 2018, doi: 10.23917/emitor.v18i01.6235.

[6] C. Fan, Z. Zhang, and D. J. Crandall, “Deepdiary: Lifelogging image captioning and summarization,” *J. Vis. Commun. Image Represent.*, vol. 55, no. March 2017, pp. 40–55, 2018, doi: 10.1016/j.jvcir.2018.05.008.

[7] X. Yang, X. He, J. Zhao, Y. Zhang, S. Zhang, and P. Xie, “COVID-CT-Dataset: A CT Image Dataset about COVID-19,” *Arxiv.Org*, vol. XX, no. Xx, pp. 1–14, 2020, [Online]. Available: https://www.medrxiv.org/.

[8] X. He *et al.*, “Sample-Efficient Deep Learning for COVID-19 Diagnosis Based on CT Scans,” *medRxiv*, vol. XX, no. Xx, p. 2020.04.13.20063941, 2020, doi: 10.1101/2020.04.13.20063941.

[9] J. P. Cohen, P. Morrison, and L. Dao, “COVID-19 Image Data Collection,” 2020, [Online]. Available: http://arxiv.org/abs/2003.11597.

[10] M. Ilyas, H. Rehman, and A. Nait-ali, “Detection of Covid-19 From Chest X-ray Images Using Artificial Intelligence: An Early Review,” pp. 1–8, 2020, [Online]. Available: http://arxiv.org/abs/2004.05436.

[11] “Covid-19 database,” *https://www.sirm.org/en/*, 2020. https://www.sirm.org/en/category/articles/covid-19-database/ (accessed Nov. 21, 2020).

[12] “Covid-19,” *https://radiopaedia.org/*, 2020. https://radiopaedia.org/search?utf8=✓&q=covid-19&scope=all&lang=us (accessed Nov. 21, 2020).

[13] “Eurorad,” *https://www.eurorad.org/*, 2020. https://www.eurorad.org/advanced-search?search=covid-19&sort_by=published_at&sort_order=ASC&page=1&filter%5B0%5D=section%3A40 (accessed Nov. 21, 2020).

[14] J. Solawetz, “Why and How to Implement Random Rotate Data Augmentation,” *https://blog.roboflow.com/*, 2020. https://blog.roboflow.com/why-and-how-to-implement-random-rotate-data-augmentation/ (accessed Nov. 27, 2021).

[15] J. Wang and L. Perez, “The Effectiveness of Data Augmentation in Image Classification using Deep Learning,” 2017.

[16] A. Howard *et al.*, “Searching for mobileNetV3,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 1314–1324, 2019, doi: 10.1109/ICCV.2019.00140.

[17] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[18] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, 2019.

[19] R. Siswosudarmo, “Tes diagnostik (Diagnostic test),” *J. Metodol. Penelit.*, p. 12, 2017, [Online]. Available: http://obgin-ugm.com/wp-content/uploads/2017/09/HRS-Kuliah-Tes-Diagnostik.pdf.

[20] C. Goutte and E. Gaussier, “Ch10_Witnesses[8463].Pdf,” no. April, 2005, doi: 10.1007/978-3-540-31865-1.

[21] B. Ramsay and E. Van Der Knaap, “Confusion Matrix-based Feature Selection Sofia Visa,” 2018.

[22] Kuliahkomputer, “Pengujian Dengan Confusion Matrix.” http://www.kuliahkomputer.com/2018/07/pengujian-dengan-confusion-matrix.html#:~:text=Confusion matrix adalah suatu metode,sebagai representasi hasil proses klasifikasi. (accessed Nov. 28, 2021).

[23] S. Setiawan, “Membicarakan Precision, Recall, dan F1-Score,” *https://medium.com*, 2020. https://stevkarta.medium.com/membicarakan-precision-recall-dan-f1-score-e96d81910354 (accessed Dec. 01, 2021).