

Optimization of 2D-CNN Setting for the Classification of Covid Disease Using Lung CT Scan

Kartika Candra Kirana, Slamet Wibawanto, Achmad Hamdan, Wahyu Nur Hidayat

Department of Electrical Engineering, Faculty of Engineering, Universitas Negeri Malang, Jl. Semarang 5, Malang 65145, Indonesia

E-mail: kartika.candra.ft@um.ac.id

Abstract

RT-PCR is considered the best diagnostic tool. Previous studies have demonstrated the reliability of CNN in classifying classifications, but CNN requires a lot of training data. Meanwhile, at the CT Scan clinic, patients are limited. Therefore, exploration of 2D-CNN settings is proposed to optimize CNN performance on limited data. We compare: (1) activation models, (2) output shapes per layer, (3) dropout layers, and (4) early stopping values. The test results show that RELU activation is better than Sigmoid. Rescaling (128x128) is better for scala (64x64) and (256x256) which affects the output shape model of each layer. In this learning stage, the use of dropouts in the CNN architecture achieves robust accuracy than the architecture that ignores dropouts. The use of 15 early stoppings is better than other values compared. 20 images of pneumonia and 20 images of covid have been tested using the proposed method and achieved 87.50% accuracy, 80.00% precision, 100% recall, and 99.89% F1-Score. Our method is superior to the the comparison method in terms of accuracy, precision, recall, and f1-score, which achieves 85%, 70%, 100%, and 82.35%, respectively.

Keywords: 2D-CNN, Covid, Lung CT Scan, classification

1. Introduction

As of mid-March 2022, there were 456,797,217 confirmed cases of COVID 19 worldwide. The death toll has increased to 6,043,094, encouraging research into the Covid classification [1]. RT-PCR is considered the best diagnostic tool, therefore several studies are exploring the method of recognizing CT scans of the lungs of Covid patients[2]–[4].

Some researchers specifically optimized convolutional neural network (CNN) algorithms as a classifier method. Pathan (2021) classified covid disease using the Binary Gray Wolf Optimizer for feature extraction and the WOABAT algorithm for CNN optimization. Their binary classification results achieve 98% binary accuracy [2].

The integration of the sparse autoencoder and feed forward neural network is also exploited to reduce the feature dimensions. That study used 1046 training images and achieved an accuracy of up to 95.7% [3].

In addition, the combination of chest X-Ray and CT images is proposed to complement the

features processed in dynamic-CNN. The training was conducted on 104,009 images and the binary accuracy test reached 92% [5]. However, the pneumonia class was ignored as in the previous two studies.

Moreover, the combination of the chest X-ray and CT image as the CNN input resulted 99,6% in binary classification and 98.28% multi-class classification. However, they also uses a large dataset of 3,877 images.[6]

A parallel bi-branch model using CNN module and Transformer module is proposed as a binary classification method to improve local and global feature extraction capability. Their results achieved 97% accuracy, but required 301 minutes of training time to 194,922 images [7].

Augmentation have been applied to 15,496 X-ray images to enhance feature representation and improve CNN performance. The result of multi-class classification reached 95.82%[8]. In other case, augmentation is also applied to reduce noise of the lung ct-scan. The combination of augmentation and CNN was applied to 1000

images and achieved a sensitivity of 80.8%, specificity of 91.5%, and F-Measure of 73.7% [9].

In addition to feature modification, evaluation of parameter functions also shows significant classification results. The use of type-based activation before softmax on CNN resulted in 99,6% accuracy. [4]. In another case study the activation model also improves the classification results[10]. They was inspiring to scrutinize the CNN settings to produce more optimal values.

Based on all those studies, CNN was successful in classification [4], [10], [11]. However, there are several 2DCNN settings that need to be set at the architectural design stage, such as the use of dropouts, output shapes, activation models, and early stopping. The contribution of this research is the comparison of attribute values that need to be set at the initialization stage so that the best value is obtained on the 2DCNN architecture for the covid-19 case study. It inspired to explore setting settings on CNN to obtain optimal classification results. Therefore, this study proposes an optimization of the 2D-CNN architecture.

2. Dataset and Method

2.1 Datasets

This research is a collaborative research between Universitas Negeri Malang and dr.Tika Health&Care Laboratory. In this laboratory, only patients diagnosed positive on PCR, treat a ct-scan to confirm the disease. Due to the needs of this laboratory, the dataset used is a binary dataset with the classes covid and pneumonia, whereas normal classes is neglected. We used the kaggle dataset [12] to observe the effect of the settings on 2D-CNN. 111 images of covid and 70 images of pneumonia were used as training data. Meanwhile, 20 images of covid and 20 images of pneumonia were used as test data. The sample data is shown in Fig.1.

2.2 Methods

In this study, setting settings on 2D-CNN is explored to obtain optimal classification results. The CNN exploratory model is shown in Fig.2.

Based on Figure 2, the CNN architecture consists of multiple convolutional layers, maximum pooling layer, flatten and dense. Convolutional layers run on offset. The purpose of image convolution is extracting features. It produces a linear transformation to the spatial information. The max-pooling layer reduce the spatial size of convolution features and reduce overfitting by providing an abstract representation of them. The flatten layer has changed features to a 1D vector. Dense layer is a function a fully

connected layer. The unit indicates that the number of nodes required for the hidden layer is the value between the number of input nodes and the output node.

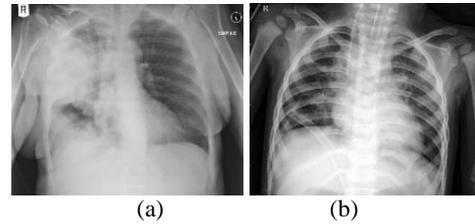


Fig. 1. Sample data (a) Covid (b) Pneumonia [12]

However, testing of 2D-CNN settings is needed to find the best model. In the first test, we evaluated the best activation model applied to each layer. There are two compared models, consist of: Sigmoid and Relu Activation. The sigmoid function is shown in Equation 1, while the Relu function is shown in Equation 2.

$$S(x) = \frac{1}{1+e^{-x}} \quad (1)$$

A feature- x in was computed by the sigmoid function- $S(x)$. it relies on the Euler's number- e .

$$f(x) = \max(0, x). \quad (2)$$

ReLU function is a piecewise linear function that returns an immediate input if positive (> 0), otherwise it returns zero.

In the second test, we compare the best output shape of each layer. The output shape affects the CNN computation, thus the results of the second trial are used to decide the best output shape that produces the most balanced combination of time and accuracy. The comparison of output shape are shown in Table 1.

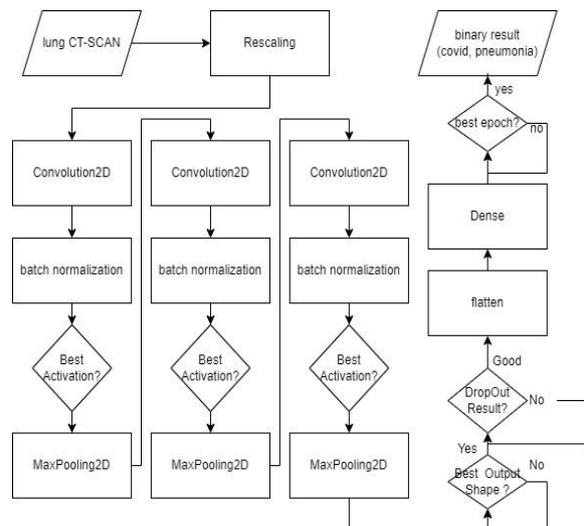


Fig. 2. The CNN exploratory model

Table 1. Output Shape

Layer	Output Share		
	1st	2st	3st
R	Output Share	Output Share	Output Share
Rescaling	(64,64,3)	(128,128,3)	(256,256,3)
Concolution2D	(64,64,8)	(128,128,16)	(256,256,32)
Max Pooling	(32,32,8)	(64,64,16)	(128,128,32)
Convolution 2D	(32,32,16)	(64,64,32)	(128,128,64)
Max Pooling	(16,16,16)	(32,32,32)	(64,64,64)
Convolution 2D	(16,16,32)	(32,32,64)	(64,64,128)
Max Pooling	(8,8,32)	(16,16,64)	(32,32,128)
Flatten	2048	16384	131072
Dense	64	128	256
Dense	3	3	3
Total Parameter	137,363	2,121,251	33,648,707

In the third test, we analyzed the use of Dropout layer. Dropout layer is used to prevent the overfitting and speed up the learning process. It refers to the removal of neurons in the form of hidden or visible layers in the network.

In the fourth test, we tested the early stopping value with variations in values {5,10,15,20,25}. The results of the fourth test are used to find the most balanced combination of time and accuracy.

2.3 Testing

System performance is measured using: accuracy, precision, recall, F1-score, and duration. Accuracy shows the ratio of the correctness of the covid and pneumonia class predictions. Recall (Sensitivity) shows the ratio of the correct Covid prediction to the original Covid data. Precision represents the ratio of the correct Covid predictions compared to the overall results predicted by Covid. F1 Score is a comparison of the average precision and recall. The four values are measured through the percentages shown in Equation 3-6. Meanwhile, duration is measured in second.

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN'} \quad (3)$$

$$Precision = \frac{TP}{TP+FP'} \quad (4)$$

$$Recall = \frac{TP}{TP+FN'} \quad (5)$$

$$F1 - Score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (6)$$

True Positive (TP) is the correct covid prediction. True Negative (TN) is the correct pneumonia prediction False Negative (FN) is a mispredicted covid class. False Positive (FP) is an mispredicted pneumonia class.

3. Result and Discussion

In the first test, activation evaluation was carried out. We compared the two best activation

functions according to Bircanoğlu et al [13], namely Sigmoid and Relu. The architectural standardization in the first test is shown in Table 2. The training data consists of 111 images of covid and 70 images of pneumonia, while the test data consists of 20 images of covid and 20 images of pneumonia.

The comparison of the activation results is shown in Fig.3. Fig. 3(a) and Fig. 3(b) is a Sigmoid activation test, while Fig. 3(c) and Fig. 3(d) is a Relu activation test. We performed tests on early stoppings 5 and 10 to validate the activation model. The using sigmoid activation reached 0.63 (63%) the maximum accuracy. Meanwhile, Relu tests showed unstable accuracy, but produced better accuracy than Sigmoid. Although the accuracy is unstable, this problem can be overcome by using the right early stopping. It can be concluded that Relu activation was better than Sigmoid, so Relu activation was set for the next experiment.

Table 2. Standardization of Output Shape on Test-1

Layer	Output Share
Rescaling	(128,128,3)
Concolution2D	(128,128,16)
Max Pooling	(64,64,16)
Convolution 2D	(64,64,32)
Max Pooling	(32,32,32)
Convolution 2D	(32,32,64)
Max Pooling	(16,16,64)
Flatten	16384
Dense	128
Dense	3
Total Parameter	2,121,251

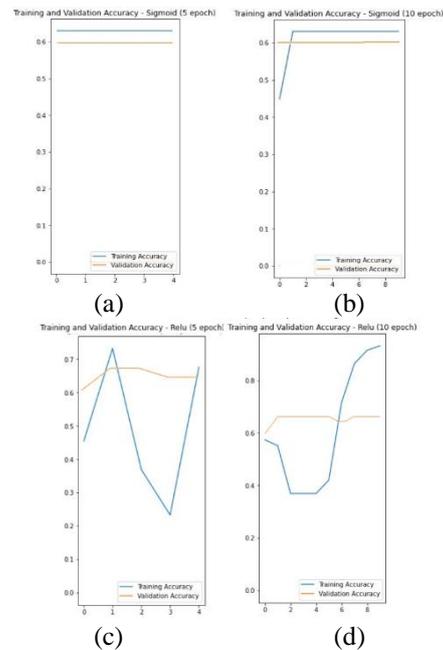


Fig. 3. Training Accuracy and Validation Based on Activation (a) Sigmoid-5 early stoppings (b) Sigmoid - 10 early stoppings (c) Relu - 5 early stoppings (d) Relu - 10 early stoppings

In the second test, we evaluated the best output shape model. The output shape model compared is shown in Table 1. We adapted the model of [4] using 128×128 . We also modify those scale by reducing with the scale of 0.5 and increasing it to a scale of 2, so there are three rescaling sets consist of: $\{256 \times 256, 128 \times 128, \text{ and } 64 \times 64\}$. Meanwhile the output shape must be adjusted to the rescaling size, so that there are three output shape shown in Table 1. The comparison results based on the output shapes are shown in Fig.4.

In Fig 4, the 3 output shape models achieve an accuracy exceeding 90%. However, Table 3 shows a significant difference in the use of 256×256 rescaling compared to 64×64 and 128×128 rescaling. The use of rescaling 64×64 and 128×128 resulted in a similar training time duration, reaching 50 s and 61 s, respectively. While rescaling 256×256 requires almost five times the training time, which is 283 s. This is triggered by the use of larger parameters. Based on the second test, the most balanced combination of time and accuracy is 128×128 . Although the use of 64×64 rescaling is shorter, the 128×128 rescaling accuracy is more stable every early stopping. Furthermore, the duration of time required is almost the same. For this reason, in further testing, 128×128 rescaling is used and the 2nd shape output model is used.

Table 3. Training Time Based on Output Shape

Testing*	Output Shape		
	1st Output Shape	2st Output Shape	3st Output Shape
Training Time	50 s	61 s	283 s
Duration	722	924 ms/step	5 s/step
Average	ms/step		

* 10 early stopping

Table 4. Training Time Based on The Use of Dropouts

Testing*	Model	
	Drop Out	Without Drop Out
Training Time	61 s	61 s
Duration	726 ms/step	924 ms/step
Average		

* 10 early stopping

Table 5. Early Stopping Testing

Early Stopping	Accuracy (%)	Training Time (s)
5	63.63	31
10	94.32	61
15	99.98	96
20	100	126
25	100	153

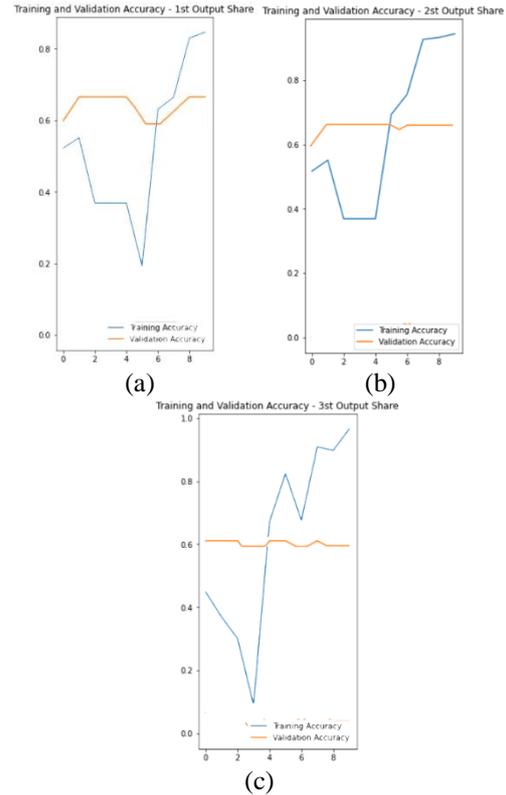


Fig. 4. Training Accuracy and Validation Accuracy Based on Output Shape (a) 64×64 (b) 128×128 (c) 256×256

In the third test, we evaluated the use of dropout and it is shown in Fig.5. Fig 5(a) and Fig 5(c) show the training stage employed dropout layers. The results show that the accuracy achieved 90.9% and 98.86% in the 10th and 15th early stoppings, respectively. While Fig 5(b) and Fig 5(d) show the training stage ignored the dropout layers. The results showed that the accuracy reached 91.4% and 98.86% in the 10th and 15th early stoppings, respectively. This indicates that the use of dropouts does not always improve accuracy results.

Table 4 is the result of the training time based on the use of dropouts. the time of both is similar, however the average per step when using dropout is faster, which is 924 ms/step. It indicated that Dropouts reduce time because discard unneeded neurons. By considering time and accuracy, it is concluded that the use of dropout is better, so that dropout is used in the next stage.

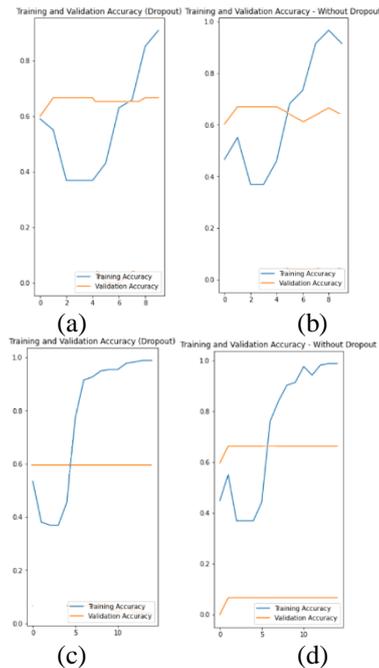


Fig. 5. Training Accuracy and Validation Based on Dropout (a)(c) Dropout (b)(d) Without Dropout (a)(b) 10 early stopping (c)(d) 15 early stopping

In test-4, the best early stopping value was tested with the set members {5,10,15,20,25}. Based on Table 5 about the results of the early stopping test, it was found that at early stoppings 20 and 25, early stoppings reached a maximum value of 100%, however the accuracy at early stopping 15 was close to the maximum value, which was 99.98%. By considering time duration and accuracy, early stopping 15 was chosen as the best early stopping.

We compared the results of testing the use of activation functions, output shapes, use of dropouts, and early stopping in the validation class after getting good accuracy in the training class, which are also shown in Fig. 3-6. The results also show that using the Relu activation function, selecting the second output shape, stopping early when reaching 15, and using dropout is superior to other value variations. However, the obtained results are in the 60-80 percent range due to the small amount of data used, namely 8 validation data.

Based on the four tests, the best architecture was obtained as shown in Fig 7. After getting the optimal setting, testing of 20 Covid data and 20 pneumonia data was carried out. The test is shown in Table 6. While the prediction output of the test data is shown in Fig.8. Based on Table 6, the proposed CNN can predict all classes of covid correctly so that it reaches 100% recall. Meanwhile, there are 15 images from 20 images of pneumonia that are classified correctly, thus achieving a precision of 80%. The precision and

recall values obtained trigger the F1-Score value to reach 99.89% and accuracy to 87.89%.

Based on the results obtained, our system can be used as a reference in initializing the 2D-CNN settings. However, there are several aspects that need to be addressed in future research, such as testing data augmentation to increase data representation in a limited dataset.

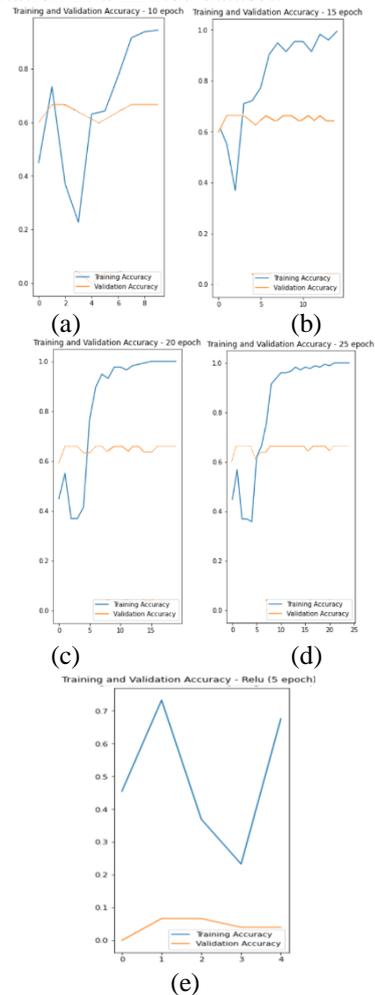


Fig. 6. Training Accuracy and Validation Based on Early stopping (a) 10 (b) 15 (c) 20 (d) 25 (e) 5

Layer (type)	Output Shape	Param #
rescaling_2 (Rescaling)	(None, 128, 128, 3)	0
conv2d_6 (Conv2D)	(None, 128, 128, 16)	448
max_pooling2d_6 (MaxPooling 2D)	(None, 64, 64, 16)	0
conv2d_7 (Conv2D)	(None, 64, 64, 32)	4640
max_pooling2d_7 (MaxPooling 2D)	(None, 32, 32, 32)	0
conv2d_8 (Conv2D)	(None, 32, 32, 64)	18496
max_pooling2d_8 (MaxPooling 2D)	(None, 16, 16, 64)	0
dropout_1 (Dropout)	(None, 16, 16, 64)	0
flatten_2 (Flatten)	(None, 16384)	0
dense_3 (Dense)	(None, 128)	2097280
dense_4 (Dense)	(None, 3)	387

=====
 Total params: 2,121,251
 Trainable params: 2,121,251
 Non-trainable params: 0

Fig. 7. CNN 2D Architecture

Table 6. Result

Testing	Result	
	The Proposed Method	The Compared Method[14]
TP	20 images	20 images
TN	15 images	14 images
FP	5 images	6images
FN	0 images	0 images
Accuracy	87.50 %	85.00 %
Precision	80.00 %	70.00 %
Recall	100 %	100 %
F1-Score	99.89%	82.35 %

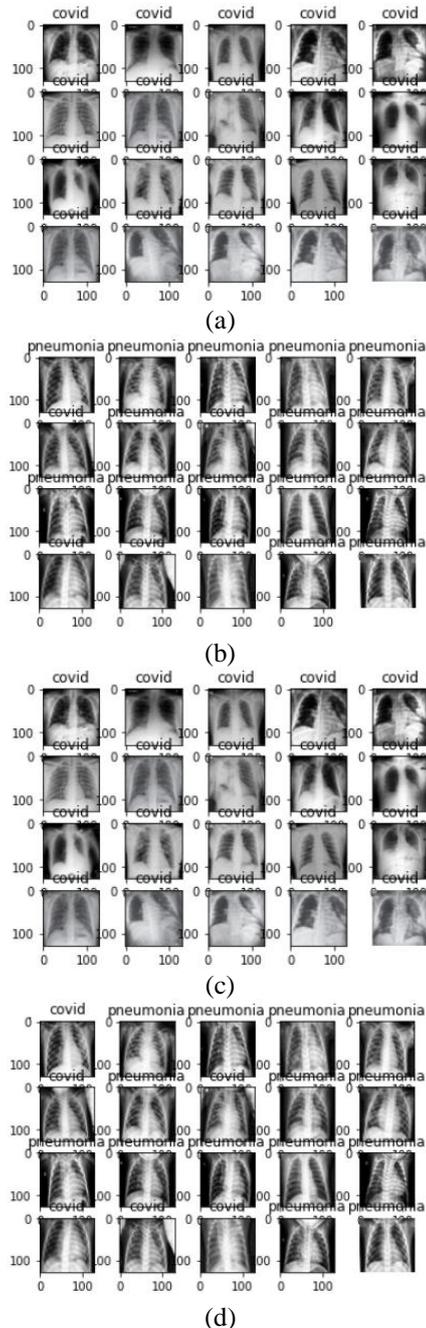


Fig. 8. Classification output (a)(c) Covid class (b)(d) Pneumonia class (a)((b) The Proposed Method (c)(d) The Compared Method

Finally, we compare the proposed method's results to those of the expert system [14]. The comparison results are also recaped in Table 6 and illustrated in Fig.8. Figures 8(b) and 8(d) show that the proposed method outperforms the expert system because the proposed method can predict one image that the compared method cannot detect accurately.

4. Conclusion

We have evaluated the settings on 2D CNN. We tested the use of activation models, output shapes, dropouts, and early stoppings. The experimental results show that relu activation is better than sigmoid. Rescaling 128×128 is better than 256×256 and 64×64 in terms of time and accuracy. Especially in this case, the use of dropout is not better. While the best early stopping obtained was 15. Based on 4 trials of setting 2D CNN, the architecture shown in Fig. 7. The training time reaches 96 seconds for 181 training images. That means each training image is computed 0.5 second on 2D-CNN. Tests on 20 images of covid and 20 images of pneumonia achieved 87.50% accuracy, 80.00% precision, 100% recall, and 99.89% F1-Score. It canbe concluded that our method is superior to the the comparison method in terms of accuracy, precision, recall, and f1-score, which achieves 85%, 70%, 100%, and 82.35%, respectively. In addition, there are several settings that can be set to optimize CNN performance, including: dropout, early stopping, output shape, and activation.

Acknowledgement

Research funded by dr tika clinic

References

- [1] WHO, "Coronavirus (COVID-19)," 2022. <https://covid19.who.int/>.
- [2] S. Pathan, P. C. Siddalingaswamy, P. Kumar, M. Pai M M, T. Ali, and U. R. Acharya, "Novel ensemble of optimized CNN and dynamic selection techniques for accurate Covid-19 screening using chest CT images," *Comput. Biol. Med.*, vol. 137, no. September, p. 104835, 2021, doi: 10.1016/j.compbiomed.2021.104835.
- [3] J. L. Gayathri, B. Abraham, M. S. Sujarani, and M. S. Nair, "A computer-aided diagnosis system for the classification of COVID-19 and non-COVID-19 pneumonia on chest X-ray images by integrating CNN with sparse autoencoder and feed forward neural network," *Comput. Biol. Med.*, vol. 141, no. November 2021, p. 105134, 2022, doi: 10.1016/j.compbiomed.2021.105134.
- [4] M. Toğaçar, N. Muzoğlu, B. Ergen, B. S. B. Yarman, and A. M. Halefoğlu, "Detection of COVID-19 findings by the local interpretable model-agnostic explanations method of types-based activations

- extracted from CNNs,” *Biomed. Signal Process. Control*, vol. 71, no. September 2021, pp. 0–3, 2022, doi: 10.1016/j.bspc.2021.103128.
- [5] G. Jia, H. K. Lam, and Y. Xu, “Classification of COVID-19 chest X-Ray and CT images using a type of dynamic CNN modification method,” *Comput. Biol. Med.*, vol. 134, no. April, p. 104425, 2021, doi: 10.1016/j.compbiomed.2021.104425.
- [6] S. Thakur and A. Kumar, “X-ray and CT-scan-based automated detection and classification of covid-19 using convolutional neural networks (CNN),” *Biomed. Signal Process. Control*, vol. 69, no. March, p. 102920, 2021, doi: 10.1016/j.bspc.2021.102920.
- [7] X. Fan, X. Feng, Y. Dong, and H. Hou, “COVID-19 CT image recognition algorithm based on transformer and CNN,” *Displays*, vol. 72, no. October 2021, p. 102150, 2022, doi: 10.1016/j.displa.2022.102150.
- [8] M. M. A. Monshi, J. Poon, V. Chung, and F. M. Monshi, “CovidXrayNet: Optimizing data augmentation and CNN hyperparameters for improved COVID-19 detection from CXR,” *Comput. Biol. Med.*, vol. 133, no. December 2020, p. 104375, 2021, doi: 10.1016/j.compbiomed.2021.104375.
- [9] M. Momeny et al., “Learning-to-augment strategy using noisy and denoised data: Improving generalizability of deep CNN for the detection of COVID-19 in X-ray images,” *Comput. Biol. Med.*, vol. 136, no. July, p. 104704, 2021, doi: 10.1016/j.compbiomed.2021.104704.
- [10] K. C. Kirana, S. Wibawanto, N. Hidayah, and G. P. Cahyono, “Improved Neural Network using Integral-RELU based Prevention Activation for Face Detection,” *Int. Conf. Electr. Electron. Inf. Eng.*, 2019.
- [11] K. C. Kirana, S. Wibawanto, N. Hidayah, and G. P. Cahyono, “The Improved Artificial Neural Network based on Cosine Similarity for Facial Emotion Recognition,” *Int. Conf. Electr. Eng. Comput. Sci. Informatics*, 2019.
- [12] P. Raikote, “Covid19-dataset.” Kaggle, 2020, [Online]. Available: <https://www.kaggle.com/datasets/pranavraikokte/covid19-image-dataset>.
- [13] C. Bircanoğlu and N. Arıca, “A comparison of activation functions in artificial neural networks,” in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 2018, pp. 1–4, doi: 10.1109/SIU.2018.8404724.
- [14] M. B. Priyantono, A. A. Rachmawan, L. A. P. Budi, and K. C. Kirana, “Sistem Prediksi Gejala Virus Korona dengan Metode Forward Chaining,” *JTERA (Jurnal Teknol. Rekayasa)*, vol. 5, no. 1, p. 111, 2020, doi: 10.31544/jtera.v5.i1.2019.111-118.