*Research Article*

# High-Performance Convolutional Neural Network Model to Identify COVID-19 in Medical Images

**Macellino Setyaji Sunarjo [1], Hong-Seng Gan [2], and De Rosal Ignatius Moses Setiadi [1,*]**

[1] Faculty of Computer Science, Dian Nuswantoro University, Semarang, Central Java 50131, Indonesia; e-mail : 111201810860@mhs.dinus.ac.id, moses@dsn.dinus.ac.id
[2] School of AI and Advanced Computing, XJTLU Entrepreneur College (Taicang), Xi'an Jiaotong -Liverpool University, Suzhou, Jiangsu, P.R. China 215400; e-mail : hongseng.gan@xjtlu.edu.cn
* Corresponding Author : De Rosal Ignatius Moses Setiadi

**Abstract:** Convolutional neural network (CNN) is a deep learning (DL) model that has significantly contributed to medical systems because it is very useful in digital image processing. However, CNN has several limitations, such as being prone to overfitting, not being properly trained if there is data duplication, and can cause unwanted results if there is an imbalance in the amount of data in each class. Data augmentation techniques are used to overcome overfitting, eliminate data duplication, and random under sampling methods to balance the amount of data in each class, to overcome these problems. In addition, if the CNN model is not designed properly, the computation is less efficient. Research has proved that data augmentation can prevent or overcome overfitting, eliminating duplicate data can make the model more stable, and balancing the amount of data makes the model unbiased and easy to learn new data as evidenced through model evaluation and testing. The results also show that the custom convolutional neural network model is the best model compared to ResNet50 and VGG19 in terms of accuracy, precision, recall, F1-score, loss performance, and computation time efficiency.

**Keywords:** Convolutional Neural Network; COVID19 detection; Image Classification; Image Recognition; Transfer learning.

**Received:** July, 10th 2023
**Revised:** August, 4th 2023
**Accepted:** August, 6th 2023
**Published:** August, 8th 2023

## 1. Introduction

Coronavirus disease 2019 (COVID-19) is an infectious disease caused by severe acute respiratory syndrome coronavirus 2, or SARS-CoV-2[1]. People with COVID-19 can spread the virus through their mouth or nose when talking, singing, breathing, coughing or sneezing [2]. Necessary to detect COVID-19 accurately because the disease has similar clinical symptoms to other infectious lung diseases. A person suspected of having contracted COVID-19 must immediately know the exact condition of their body so that they can isolate themselves, receive treatment, and warn other people who have had direct contact [3].

Research proves that X-rays and computed tomography scans (CT scans) help reveal anomalies that indicate COVID-19 disease[4]. X-rays and CT scans play an important role in diagnosing COVID-19 disease, especially when clinical trials are not available [5]. However, CT scans have a long process of producing images, expensive scanners are not available in underdeveloped countries. They cannot be used for pregnant women and children due to the high radiation hazard[6]. Conversely, X-ray images are the fastest and easiest form of imaging to obtain with lower side effects on the human body [1].

Artificial intelligence has proven effective in recent years and plays an important role in various medical fields. Artificial intelligence can help doctors diagnose diseases more accurately[7]. Radiologists suggest that artificial intelligence is useful in detecting COVID-19 using X-ray images. The contribution of artificial intelligence in medical imaging as the main source of information has been confirmed by many experts. It serves as a tool for early diagnosis in describing complications of COVID-19[6]. Artificial intelligence, especially machine learning and deep learning, is gaining popularity in the medical field because it makes it possible to classify images without human intervention. [8].

DOI : 10.33633/jcta.v1i1.8936

Machine learning (ML) is a term used to describe the concept of software that learns automatically to solve problems or perform tasks. ML becomes more and more accurate over time through the learning process. ML works by receiving training data as input, building a mathematical model based on the input, and then using the mathematical model to solve the problem.[9]. ML has several techniques, such as k-nearest neighbors (KNN), support vector machine(SVM), and random forest(RF)[10]–[12]. Several ML techniques have been used to diagnose COVID-19. However, these techniques have the same weaknesses, namely low diagnostic accuracy, long prediction time, and high complexity [9]. Feature selection also has a large influence on machine learning performance. Biased selection of features can lead to discrimination between classes [13].

Deep learning (DL) is a part of ML that is inspired by information processing patterns in the human brain. DL does not require rules designed by humans to operate, instead, it uses large amounts of data to map inputs onto specific labels. DL is designed using many layers of artificial neural network algorithms. Each layer provides a different interpretation of the input data. Unlike machine learning, deep learning can automate feature selection learning[13], [14]. DL can learn complicated features accurately from large data to provide better classification performance [1]. DL cannot be adequately trained and get high accuracy if there is an imbalance in the data [8]. DL has several models, such as a recursive neural network (RvNN), recurrent neural network (RNN), and convolutional neural network (CNN)[13]. The spread of COVID-19 that spread very quickly triggered extensive research on developing various deep learning models to help diagnose COVID-19 disease that does not require manual feature engineering, so deep learning algorithms are very important to achieve promising performance in detecting COVID-19[8].

The RvNN can achieve predictions with a hierarchical structure and produce output using composition vectors. RvNN produces an architecture for processing objects with random structures, such as graphs or trees. This approach produces a distributed representation of a recursive data structure. The network is trained using backpropagation through a structure learning system (BPTS), which can support a tree-like structure. RvNN is especially effective in natural language processing (NLP) contexts [13]. The RNN is a model commonly used in DL. RNN uses sequential data in the network. Structures embedded in data sequences provide valuable information and are important for many applications. Thus, recurrent neural networks can be classified as part of short-term memory. The main application of recurrent neural networks is in the field of speech processing and natural language processing contexts[13].

CNN is a deep learning model that implements a convolution process, where the convolution process is a mathematical operation between the input data and a set of filters that can be learned to create a feature map. Feature maps extract local features, such as edges, sharp corners, shapes, and gradients, in solving various classification problems[15]. Convolutional neural networks have significantly contributed to medical systems because they are very useful in digital image processing [8]. CNN shows superiority in digital image processing, making it very useful in medical diagnosis. Convolutional neural networks make it possible to perform advanced image classification without feature engineering[8]. Convolutional neural networks have a weight division feature to reduce the number of trainable network parameters. The weight-sharing feature can help networks to improve generalization and avoid overfitting. Convolutional neural networks facilitate the implementation of large-scale networks compared to other methods[13].

CNNs customized for specific tasks can have the same or even better performance than transfer learning. The research results have proven that transfer learning from different domains does not significantly affect medical imaging performance, so custom convolutional neural networks are chosen as the main method in medical imaging compared to transfer learning models, such as ResNet and VGG[13]. ResNet has a long training time, and VGG uses a lot of memory, which is another reason for CNN customization compared to ResNet and VGG [16].

CNN has several limitations, such as being prone to overfitting, not being adequately trained if there is duplication of data, and can cause unwanted results if there is an imbalance in the amount of data in each class. Overfitting is the main problem of the CNN model in obtaining good generalization [13]. Training a CNN model that contains a duplication of data and an unequal amount of data in each class makes the model unable to be appropriately trained and can lead to unwanted results. High accuracy cannot guarantee the effectiveness of COVID-19 detection if there is a duplication of data and an imbalance in the amount of data in each class[8].

Based on some of the literature above, this study has the objectives to:

1. Customise the CNN model for high-performance and efficient detection of COVID-19.
2. Build new datasets by compiling various datasets, then eliminating duplication with the MD5 hashing technique.
3. Apply data augmentation random undersampling technique to balance the amount of data to overcome overfitting.
4. Comparing the proposed CNN model with the ResNet50 and VGG19 models based on the accuracy, loss, precision, recall, and F1-score values.

The rest of the paper is presented in five sections, namely: a literature review that discusses the related methods and the reasons for selecting the method; a proposed method that provides illustrations and discussion of the proposed method; results and discussion that discusses how to implement and compile datasets, classify and study ablation; the comparison that compares the performance of the proposed method with popular transfer learning such as VGG19 and RestNet50, and finally as a conclusion.

## 2. Literature Review

CNN is widely used in image processing because it can handle classification and image recognition problems well and greatly increases accuracy. CNN is a powerful and universal DL model[8]. In a real-life scenario, CNN effectively detects COVID-19 because it has a more effective sample collection process than a nasal swab and can diagnose quickly[15].

The DLs take up a lot of time and data, so training large DL models from scratch is generally not a good idea. A method has been developed to overcome deep learning problems, namely transfer learning. Transfer learning is reusing previously trained networks and transferring the learned model into a new one. Additional training data and modified neural layers can also be incorporated into the new model[8]. Transfer learning has at least two pre-training models, namely ResNet and Visual Geometry Group (VGG)[1].

ResNet is the most well-known pre-training model that has been used extensively in the COVID-19 classification process. ResNet consists of several residual blocks, where the output of each convolution block is added to the output of the convolution block at a deeper stage. ResNet has been widely used in detecting COVID-19 using X-ray images[1].

The VGG won the 2014 ImageNet Large Scale Visual Recognition Challenge (ILSVRC2014). Visual Geometry Group is a model with a simple architecture but still performs effectively. The Visual Geometry Group has two types of architectures, namely VGG16 and VGG19. VGG16 consists of sixteen convolution layers, while VGG19 consists of nineteen convolution layers. The Visual Geometry Group has five convolution blocks using a 3x3 fixed kernel, where the first two blocks each consist of two convolution operations, and the last three blocks each consist of three convolution operations[1].

It is assumed that the model training performance can be increased by adding more convolution blocks. However, in practice, models at deeper layers have reduced performance and often return reduced results compared to less deep models. This problem occurs due to the disappearing gradient. ResNet solves this problem by passing through multiple layers and feeding output from one layer to the next[1]. However, ResNet has a long training time so it has limited applicability [16].

VGG uses small filters with the same efficiency level as large filters. Using small filters reduces computational complications by reducing the number of parameters. In general, VGG obtains significant results for image classification problems. VGG has enlarged depth, homogeneous topology and simplicity. However, VGG has a very high computational cost because it contains approximately 140 million parameters[13]. VGG also requires large memory and high-configuration hardware [16]. In research [17], VGG19 has been tested for the identification of COVID-19, and this model is superior to VGG16. Likewise, the

ResNet50 model also has reliable performance in research [18] for the detection of COVID-19, the model proposed in this study is also compared with the two models.

## 3. Proposed Method

This research begins with how to build a new dataset that comes from compiling various datasets. This maximizes the machine learning process because CNN requires a large dataset to improve its performance. Several dataset compilation steps are carried out by combining datasets, eliminating duplication, under-sampling, resizing images, normalizing, splitting data and augmenting the training data. Fig. 1 shows an illustration of the stages of compiling a new dataset, and then this dataset is used to test the proposed model.

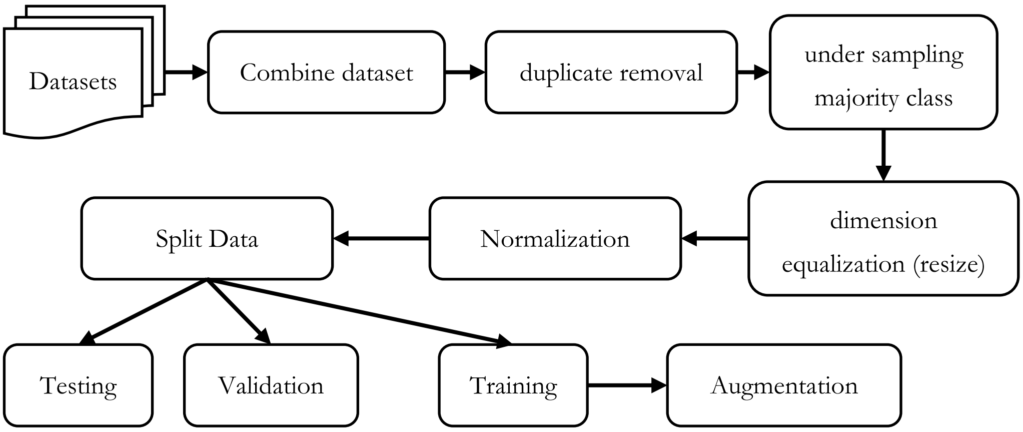


**Figure 1.** New dataset compilation process.

The CNN model is built using an instance of the Sequential class owned by Keras. Each layer is defined into a Sequential class. This study uses four convolutional layers with the ReLU activation function, four pooling layers, one fully connected layer, and one output layer with Sigmoid as a classifier. Before entering into the fully connected layer, the input must be changed first through the flatten layer. The convolutional layer functions to extract the attributes in the image. The pooling layer functions to reduce image resolution so that the model training process becomes faster. The flatten layer functions to flatten the input, changing the input from a two-dimensional matrix to a one-dimensional array. The fully connected layer is the layer where all the neurons from the previous layer are connected to the neurons in the next layer. This study uses a dense layer with the ReLU activation function as the fully connected layer. The ReLU function has an output value of zero if the input is negative, otherwise, one if the input is positive. The output layer is created using the number of units that adjusts the number of labels in the dataset. Fig. 2 illustrates the construction diagram of the proposed custom convolutional neural network model.

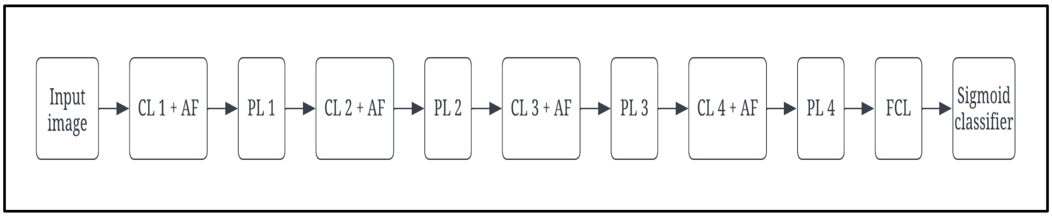
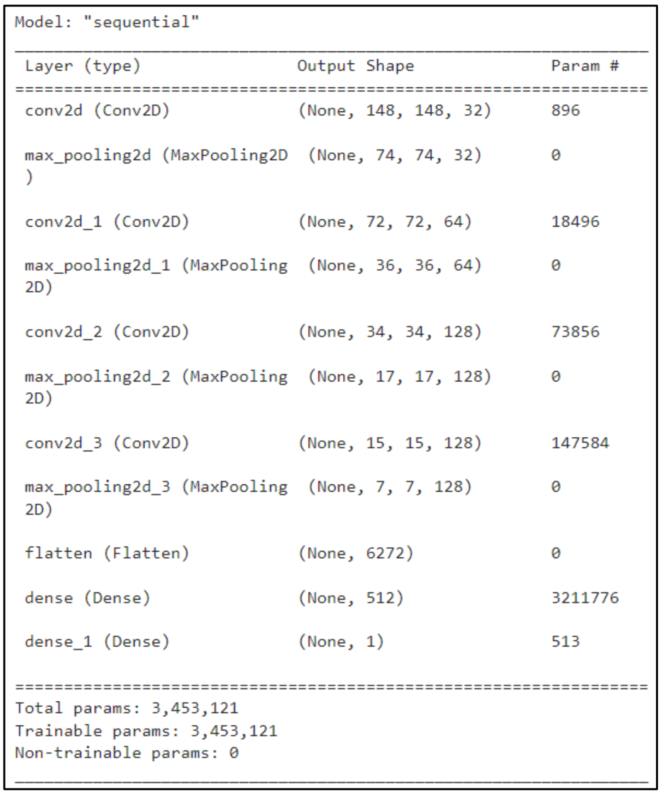


**Figure 1.** Proposed Model CNN.

where CL is convolution layer, PL is pooling layer, AF is activation function, FCL is fully connected later. Meanwhile, to see more clearly about the hyperparameters and optimizer used, you can see Figure 2.



```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 148, 148, 32)      896

 max_pooling2d (MaxPooling2D  (None, 74, 74, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 72, 72, 64)        18496

 max_pooling2d_1 (MaxPooling  (None, 36, 36, 64)       0
 2D)

 conv2d_2 (Conv2D)           (None, 34, 34, 128)       73856

 max_pooling2d_2 (MaxPooling  (None, 17, 17, 128)      0
 2D)

 conv2d_3 (Conv2D)           (None, 15, 15, 128)       147584

 max_pooling2d_3 (MaxPooling  (None, 7, 7, 128)        0
 2D)

 flatten (Flatten)           (None, 6272)              0

 dense (Dense)               (None, 512)               3211776

 dense_1 (Dense)             (None, 1)                 513

=================================================================
Total params: 3,453,121
Trainable params: 3,453,121
Non-trainable params: 0
_________________________________________________________________
```

**Figure 1.** Hyperparameters and output shape of the proposed CNN model.

The case in this study is a binary classification so that the output produced by the model is a single number zero and one. Thus, the output layer has neurons worth one. The Output Shape column contains information about the output size produced by each layer. The previously defined input image sizes are (150, 150, 3). CL 1 uses a filter of 32, kernel size (3, 3), and stride has default size (1, 1), so that each image will produce 32 output images with size (148, 148). Furthermore, the image's resolution is reduced while maintaining the information in the image using the PL size (2, 2) and the default stride (None) so that the stride value will follow the PL size. The output images from PL 1 are 32 images of size (74, 74).

The CL 2 uses a filter of 64, kernel (3, 3), and stride (1, 1) so that each image produces 64 output images of size (72, 72). Furthermore, the image resolution is reduced while maintaining the information in the image through PL 2 (2, 2) and stride (2, 2), resulting in 64 output images (36, 36). CL 3 uses a filter of 128, kernel (3, 3), and stride (1, 1), so that each image produces 128 output images of size (34, 34). Image resolution is reduced again while maintaining image information through PL 3 (2, 2) and stride (2, 2) so that it will produce 128 output images (17, 17). CL 4 uses a filter of 128, kernel (3, 3), and stride (1, 1), so that each image produces 128 output images (15, 15). Furthermore, the image resolution is reduced while maintaining the information in the image via PL 4 (2, 2) and stride (2, 2) so that it will produce 128 output images of size (7, 7) because PL rounds down.

The final output is 128 images (7, 7), then converted into a one-dimensional array, producing an output of 6,272 sizes via a flatten layer. The output enters the FCL, which has 512 neurons so that it will produce an output of size (512). Then, the output enters the output layer with one neuron to produce a new size output (1). The output of the last layer will be used as the final model for the binary classification case.

## 4. Results and Discussion

### 4.1. Dataset Compilation

The research is implemented with Python programming using several libraries, modules, APIs, and frameworks in building models. The libraries used include Google Collab with the drive class, NumPy, Matplotlib, and OpenCV with the imread class. While the modules used are OS, zipfile with the ZipFile class, shutil, random, hashlib, and math. Keras with ImageDataGenerator, Sequential, Conv2D, Max-Pooling2D, Flatten, Dense, Input, ResNet50, and VGG19 classes is the API used in this research. Scikit-learn with class classification_report is the framework used.

This study compiles nine COVID-19 datasets, viz [19]–[27]. Several datasets have a different number of classes, while in this study only carried out binary classification. Then it must be processed manually first to facilitate the data preprocessing stage, as follows:

1. The dataset [19] contains four subfolders: COVID-19, Lung_Opacity, Normal, and Viral Pneumonia. Only two folders are used, namely COVID-19 from the Institute for Diagnostic and Interventional Radiology and Hannover Medical School for classes and the Normal class from the Radiological Society of North America (RSNA).
2. The dataset [20] is from Guangzhou Women and Children's Medical Center, containing chest X-ray images of pediatric patients aged one to five years as part of routine clinical care. This dataset contains two subfolders, namely covid and normal, so it can be used immediately.
3. The dataset [21] is from Eurorad, Radiopaedia, and coronacases.org and is the same source as the dataset [20]. This dataset contains three folders, namely COVID, normal, and pneumonia. Then just select the COVID and normal folder.
4. The dataset [22] is from the University of Montreal, contains two subfolders, namely test and train. Each subfolder contains three subfolders each, namely Covid, Normal, and Viral Pneumonia. Then the Covid and Normal folders are selected. The Covid and Normal subfolders from the test and train subfolders are merged into one.
5. The dataset [23] comes from the Kaggle web, and contains three subfolders, namely COVID-19, normal, and pneumonia, so the COVID-19 and normal subfolders are selected.
6. The dataset [24] comes from a research team from Qatar University and the University of Dhaka in collaboration with doctors and medical specialists from Pakistan and Malaysia, the same source as the dataset[20], as well as a research team from the University of Waterloo, City of London, National Research Council Canada (NRC), and Selayang Hospital. This dataset contains three subfolders, namely COVID-19, normal, and pneumonia, so the COVID-19 and normal subfolders were selected.
7. The dataset [25] comes from various universities and doctors approved by the University of Montreal and contains two subfolders, namely Train and Val. Each subfolder contains three subfolders respectively, namely Covid, Normal, and Pneumonia. The Covid and Normal subfolders were merged, and the Train and Val folders were merged.
8. The dataset [27] comes from the same source as datasets[24] and [25], and contains three folders, namely test, train, and val. Each folder contains four subfolders respectively, namely COVID-19, normal, pneumonia, and tuberculosis. The cases in this study do not use a dataset that has been divided into training data, validation data and test data. The cases in this study are a binary classification so they only need the COVID-19 and normal subfolders. The COVID-19 and normal subfolders from the test, train, and val folders were merged into one and moved into a new folder.
9. The dataset[26] comes from the same source as [19], [27], as well as CheXpert. Contains two subfolders, namely 3-classes and 5-classes. The 3-classes subfolder contains three subfolders, namely Test, Train, and Val. Each subfolder contains three subfolders, namely Covid-19, Normal, and Pneumonia. The 5-classes subfolder contains three subfolders, namely Test, Train, and Val. Each subfolder contains five subfolders, namely Bacterial, Covid-19, Lung Opacity, Normal, and Viral. The Covid-19 and Normal subfolders were selected and used for this study.

Based on the compilation of the nine datasets above, the graph shown in Figure 3 is produced. Meanwhile, the compiled sample dataset images are presented in Fig. 4. Based on the data in the graph, there are 8,950 medical images of COVID-19 and 15,697 normal images. This shows that the class is not balanced, so the initial processing is done, namely duplication search and undersampling.

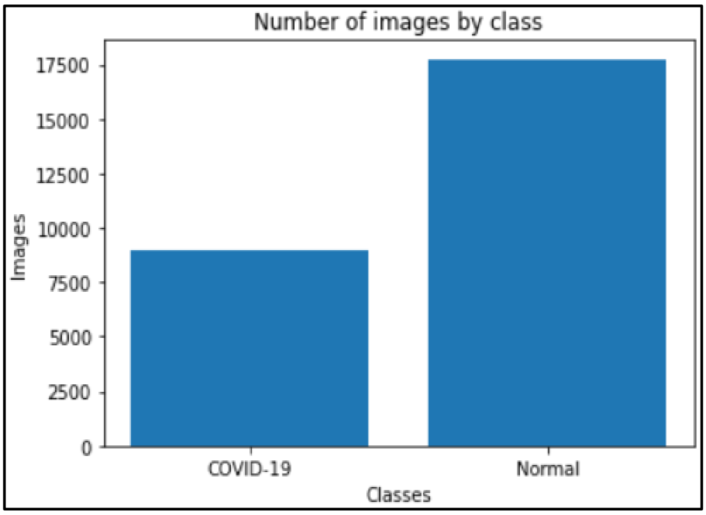


**Figure 3.** Number of images comparison for each class before preprocesing.

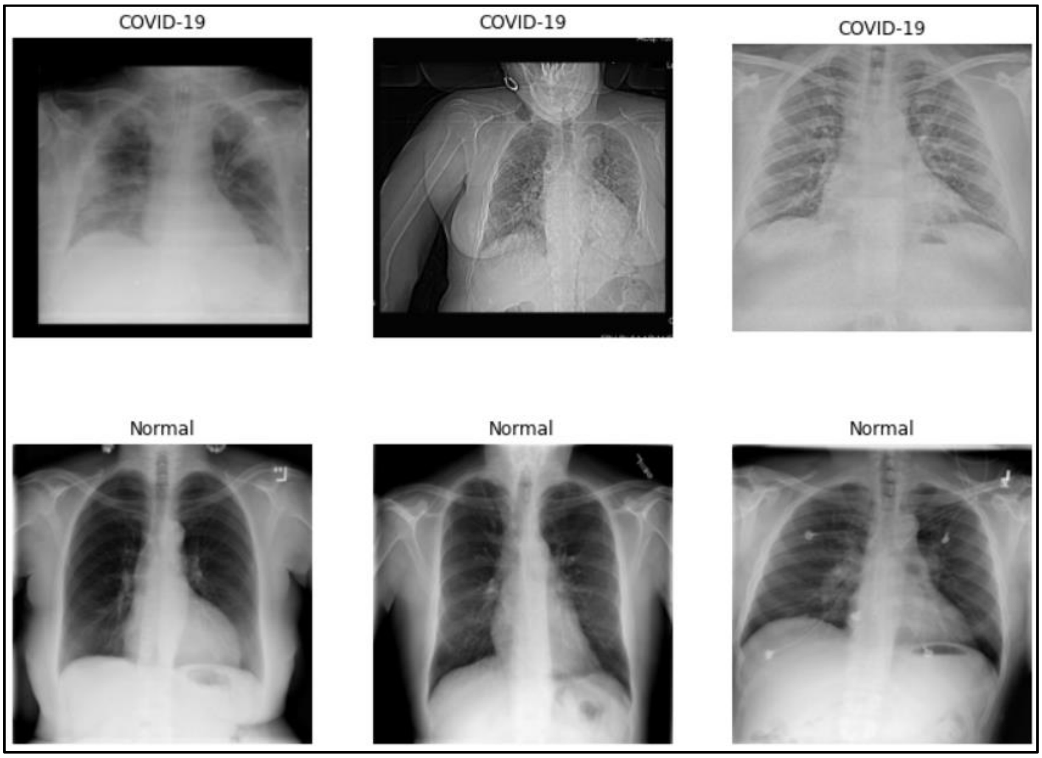


**Figure 4.** Sample x-ray images of COVID-19 and Normal Classes.

The dataset collected consists of a compilation of nine datasets, which includes duplicate data. To identify duplicate images, the MD5 hash function is employed. A glimpse of the search results can be seen in Figure 5. The findings reveal that duplication is exclusive to the normal class, with no instances found within the COVID-19 class. An under-sampling approach was executed, equalizing the number of COVID-19 images to 8950 images. Furthermore, resizing is carried out to equalize the image's dimensions because knowing the largest and smallest dimensions is necessary. Where the largest dimension is 5623×4757, and the smallest dimension is 157×156, then the entire image is resized to 150×150 pixels. It is generally a better practice to reduce the dimensions of a large image to a smaller dimension than to increase the size of a small image to a larger one. This is because enlarging the thumbnail stretches the pixels in the thumbnail, which can obscure the model's ability to learn key features, such as object boundaries.

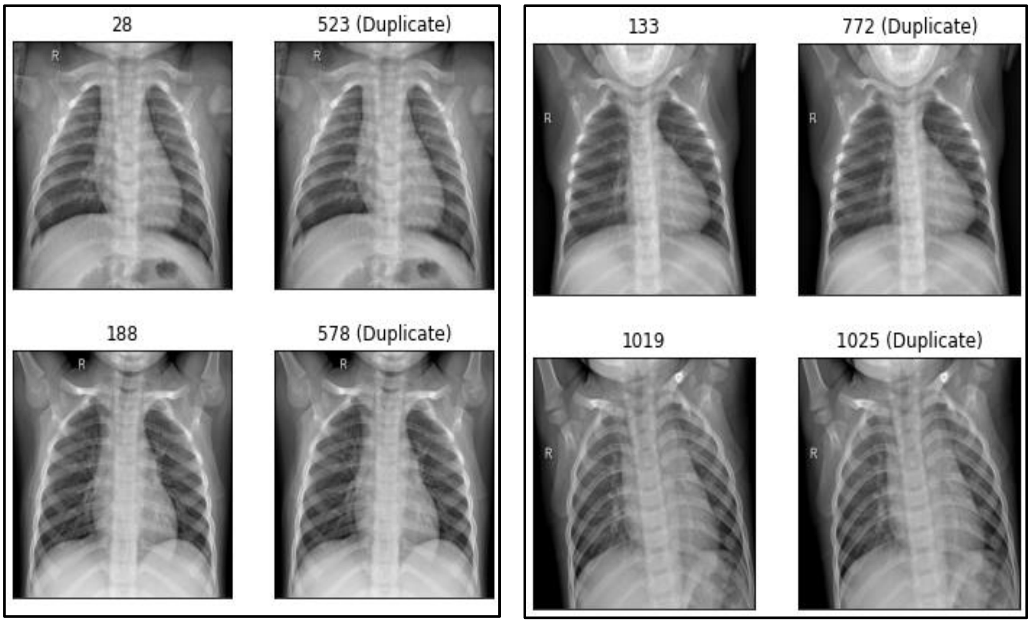


**Figure 5.** Sample duplicate image search results

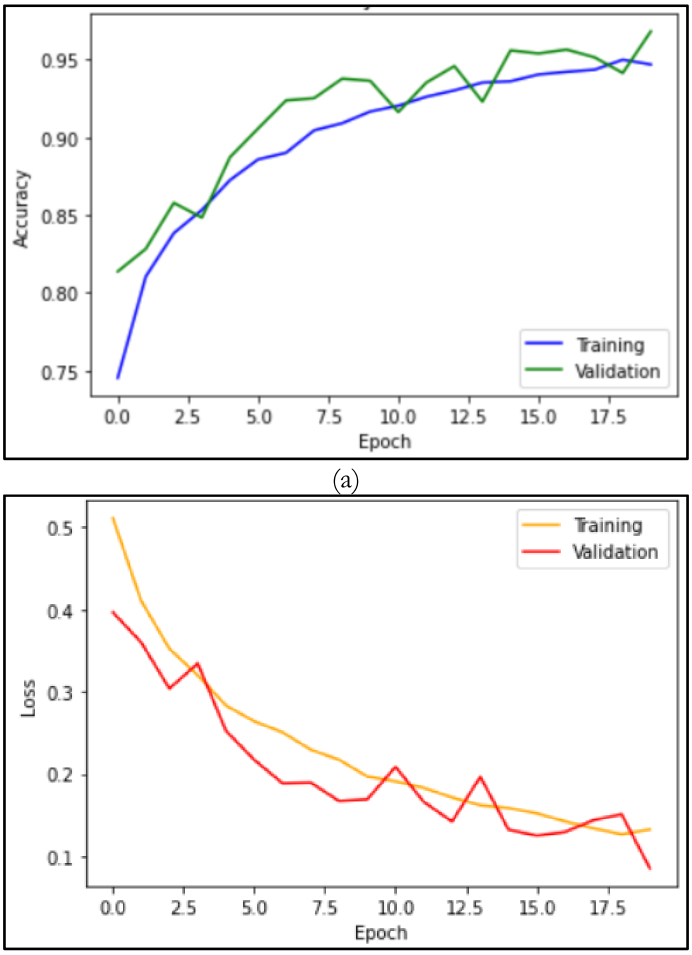
The research divides the dataset with a ratio of 60:20:20 for each training data, validation data, and test data. Because each class has 8,950 images, there are 17,900 images in total. So that the number of training data in the study was 10,740 images, while the number of validation and test data was 3,580, respectively. Normalization is done by dividing all pixels by 255 so that the pixel value of each image is between zero and one. Normalization is performed on training, validation, and test data because neural network models produce better results when the input is normalized. The normalization technique using this method is equivalent to using MinMaxScaler. MinMaxScaler uses Eq. (1) to perform normalization.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

The final step in dataset compilation is augmentation. This is done by making variations of the training data to increase the variety of data used in model training. The augmentation process creates variations from the data that are slightly different from the original data so as to avoid overfitting. The technique of rotating, tilting, enlarging, and horizontally inverting the image was chosen in this study. The augmentation in this study was not done offline but by transforming the data during the training process so that the amount of training data does not change.

### 4.2. Classification

The proposed CNN model that has been described in section 3.2 is used in this study. Before using it, the model configuration for training was carried out using the Adam optimizer, binary cross-entropy loss function, and accuracy metrics. We use the default batch of 32, so the number of batches to be executed in each epoch for training data is 336. Meanwhile, for validation data is 112 for each epoch. The results of the accuracy and loss plots of the proposed model are presented in Figure 6. Based on Figure 6, it appears that an accuracy of 0.9628 and a loss of 0.1000 are obtained in the training data. While precision, recall and validation are presented in Table 1.



(a)

(b)

**Figure 6.** Plot results of proposed model {(a) accuracy; (b)loss}.

**Table 1.** Precision, recall and validation results.

| Measurement | COVID-19 | Normal |
|---|---|---|
| Precision | 0.96 | 0.97 |
| Recall | 0.97 | 0.95 |
| F1-score | 0.96 | 0.96 |

### 4.3. Ablation Study

In this study, we also conducted an ablation study to eliminate the duplication removal process. This is to prove that data duplication can reduce classification performance. In this study, five experiments were carried out, and the results are presented in Table. 2. Based on the experiment, the model works much better and is stable without data duplication. Thus, data duplication must be eliminated to maximize the model training process.

**Table 2.** Comparison performance on dataset with and without duplication removal.

| Number of Testing | Without duplication removal | | With duplication removal | |
|---|---|---|---|---|
| | Accuracy | Loss | Accuracy | Loss |
| Test 1 | 0.9318 | 0.2264 | 0.9631 | 0.1088 |
| Test 2 | 0.9517 | 0.1417 | 0.9659 | 0.1033 |
| Test 3 | 0.8964 | 0.2685 | 0.9670 | 0.0882 |
| Test 4 | 0.9545 | 0.1311 | 0.9668 | 0.1014 |
| Test 5 | 0.9441 | 0.1638 | 0.9662 | 0.0924 |
| Average | 0.9357 | 0.1863 | 0.9658 | 0.09882 |

## 5. Comparison

In this study, a comparison was also made with the transfer learning model, namely, ResNet50 and VGG19. For the ResNet50 and VGG19 models to work properly in this study, fine-tuning was done by adding a fully connected layer using the ReLU activation function and an output layer using Sigmoid as a classifier. The results of comparing accuracy and loss are presented in Figure 7, which shows that the proposed model has the best performance, followed by VGG19 and ResNet50. We also measure the computing speed with the same hardware and software. It is shown that the proposed model has a faster computing time, as in Figure 8.

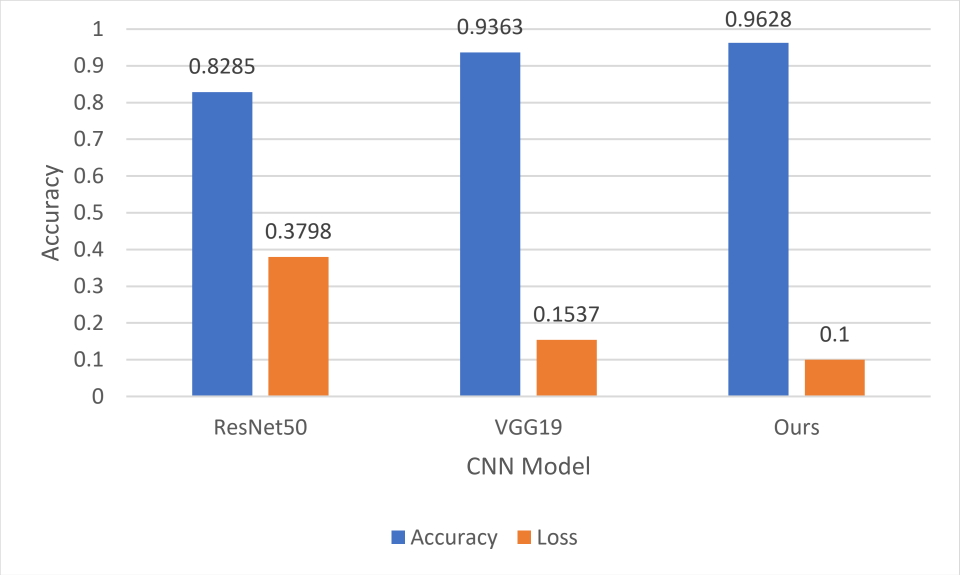


**Figure 7.** Performance comparison based on Accuracy and Loss.

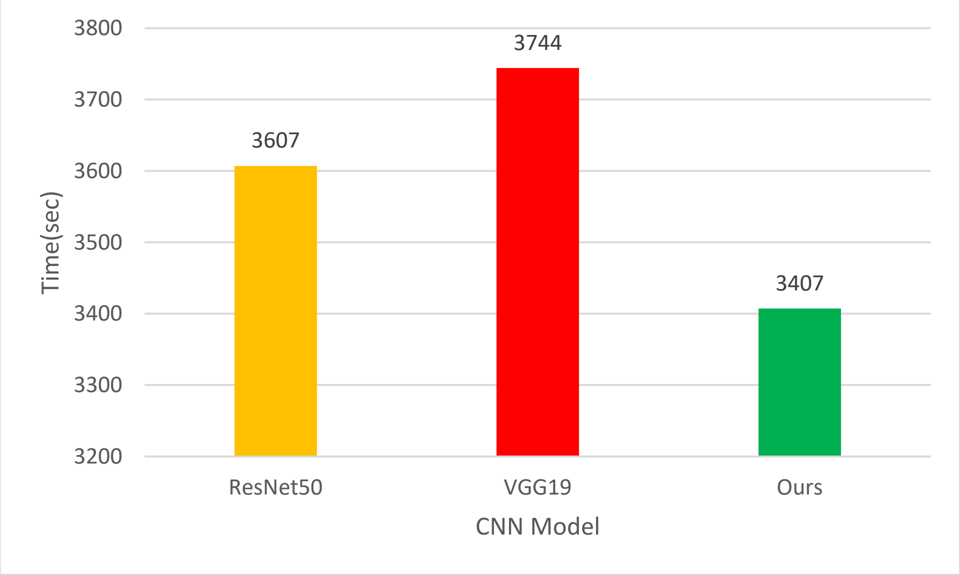


**Figure 8.** Performance comparison based on computation time.

## 6. Conclusions

This research has successfully built a CNN model with 11 layers for identifying COVID-19 on a compilation of nine medical image datasets. The compilation of the nine datasets has carried out several preprocesses, such as removing duplicates with the MD5 hash function, resizing, sampling and balancing, and normalizing so that a complete dataset is obtained for learning. This model has high performance with a validation accuracy of 96.28%. These results improve the performance of popular transfer learning models such as VGG19 and ResNet50. In addition, the computational time of the proposed model is also more efficient than both models.

**Author Contributions:** Conceptualization: M.S. Sunarjo and D.R.I.M, Setiadi; methodology: M.S. Sunarjo and D.R.I.M, Setiadi; software: M.S. Sunarjo.; validation: M.S. Sunarjo and D.R.I.M, Setiadi; D.R.I.M, Setiadi.; formal analysis, M.S. Sunarjo; investigation, X.X.; resources, M.S. Sunarjo; data curation: M.S. Sunarjo; writing—original draft preparation: M.S. Sunarjo; writing—review and editing: D.R.I.M, Setiadi and H.S. Gan; visualization: M.S. Sunarjo; supervision: D.R.I.M, Setiadi and H.S. Gan; project administration: D.R.I.M, Setiadi and H.S. Gan.

**Funding:** This research received no external funding

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

[1] P. Aggarwal, N. K. Mishra, B. Fatimah, P. Singh, A. Gupta, and S. D. Joshi, “COVID-19 image classification using deep learning: Advances, challenges and opportunities,” *Comput. Biol. Med.*, vol. 144, p. 105350, May 2022, doi: 10.1016/j.compbiomed.2022.105350.

[2] World Health Organization, “WHO Coronavirus (COVID-19) Dashboard,” 2022. https://covid19.who.int/ (accessed Jul. 19, 2022).

[3] A. H. Barshooi and A. Amirkhani, “A novel data augmentation based on Gabor filter and convolutional deep learning for improving the classification of COVID-19 chest X-Ray images,” *Biomed. Signal Process. Control*, vol. 72, p. 103326, Feb. 2022, doi: 10.1016/j.bspc.2021.103326.

[4] O. S. Albahri *et al.*, “Systematic review of artificial intelligence techniques in the detection and classification of COVID-19 medical images in terms of evaluation and benchmarking: Taxonomy analysis, challenges, future solutions and methodological aspects,” *J. Infect. Public Health*, vol. 13, no. 10, pp. 1381–1396, Oct. 2020, doi: 10.1016/j.jiph.2020.06.028.

[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, p. 104425, Jul. 2021, doi: 10.1016/j.compbiomed.2021.104425.

[6] A. Gopatoti and P. Vijayalakshmi, “CXGNet: A tri-phase chest X-ray image classification for COVID-19 diagnosis using deep CNN with enhanced grey-wolf optimizer,” *Biomed. Signal Process. Control*, vol. 77, p. 103860, Aug. 2022, doi: 10.1016/j.bspc.2022.103860.

[7] L. Kong and J. Cheng, “Classification and detection of COVID-19 X-Ray images based on DenseNet and VGG16 feature fusion,” *Biomed. Signal Process. Control*, vol. 77, p. 103772, Aug. 2022, doi: 10.1016/j.bspc.2022.103772.

[8] N. A. Baghdadi, A. Malki, S. F. Abdelaliem, H. Magdy Balaha, M. Badawy, and M. Elhosseini, “An automated diagnosis and classification of COVID-19 from chest CT images using a transfer learning-based convolutional neural network,” *Comput. Biol. Med.*, vol. 144, p. 105383, May 2022, doi: 10.1016/j.compbiomed.2022.105383.

[9] W. M. Shaban, A. H. Rabie, A. I. Saleh, and M. A. Abo-Elsoud, “Accurate detection of COVID-19 patients based on distance biased Naïve Bayes (DBNB) classification strategy,” *Pattern Recognit.*, vol. 119, p. 108110, Nov. 2021, doi: 10.1016/j.patcog.2021.108110.

[10] S. Mohammed *et al.*, “A comparative analysis of data mining techniques for agricultural and hydrological drought prediction in the eastern Mediterranean,” *Comput. Electron. Agric.*, vol. 197, p. 106925, Jun. 2022, doi: 10.1016/j.compag.2022.106925.

[11] D. R. Ignatius Moses Setiadi *et al.*, “Comparison of SVM, KNN, and NB Classifier for Genre Music Classification based on Metadata,” in *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Sep. 2020, pp. 12–16. doi: 10.1109/iSemantic50169.2020.9234199.

[12] F. P. Pramadi, C. Atika Sari, E. H. Rachmawanto, and D. Rosal Ignatius Moses Setiadi, “Flowers Identification using First-order Feature Extraction and Multi-SVM Classifier,” in *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Sep. 2020, pp. 22–27. doi: 10.1109/iSemantic50169.2020.9234260.

[13] L. Alzubaidi *et al.*, “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions,” *J. Big Data*, vol. 8, no. 1, p. 53, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[14] A. Susanto, I. U. Wahyu Mulyono, C. Atika Sari, E. Hari Rachmawanto, D. R. I. M. Setiadi, and M. K. Sarker, “Handwritten Javanese script recognition method based 12-layers deep convolutional neural network and data augmentation,” *IAES Int. J. Artif. Intell.*, vol. 12, no. 3, p. 1448, Sep. 2023, doi: 10.11591/ijai.v12.i3.pp1448-1458.

[15] A. Sharma, K. Singh, and D. Koundal, “A novel fusion based convolutional neural network approach for classification of COVID-19 from chest X-ray images,” *Biomed. Signal Process. Control*, vol. 77, p. 103778, Aug. 2022, doi: 10.1016/j.bspc.2022.103778.

[16] Y. Zhou, H. Chang, Y. Lu, X. Lu, and R. Zhou, “Improving the Performance of VGG Through Different Granularity Feature Combinations,” *IEEE Access*, vol. 9, pp. 26208–26220, 2021, doi: 10.1109/ACCESS.2020.3031908.

[17] D. Arias-Garzón *et al.*, "COVID-19 detection in X-ray images using convolutional neural networks," *Mach. Learn. with Appl.*, vol. 6, no. August, p. 100138, 2021, doi: 10.1016/j.mlwa.2021.100138.

[18] S. Rajpal, N. Lakhyani, A. K. Singh, R. Kohli, and N. Kumar, "Using handpicked features in conjunction with ResNet-50 for improved detection of COVID-19 from chest X-ray images," *Chaos, Solitons and Fractals*, vol. 145, p. 110749, 2021, doi: 10.1016/j.chaos.2021.110749.

[19] P. Viradiya, "COVID-19 Radiography Dataset," *Kaggle*, 2021. https://www.kaggle.com/datasets/preetviradiya/covid19-radiography-dataset (accessed Nov. 18, 2021).

[20] A. Rahman, "COVID-19 Chest X-ray Image Dataset," *Kaggle*, 2020. https://www.kaggle.com/datasets/alifrahman/covid19-chest-xray-image-dataset (accessed Nov. 18, 2021).

[21] S. Kumar and S. Shastri, "COVID19+PNEUMONIA+NORMAL Chest X-Ray Image Dataset," *Kaggle*, 2020. https://www.kaggle.com/datasets/sachinkumar413/covid-pneumonia-normal-chest-xray-images (accessed Nov. 18, 2021).

[22] Suddirutten, "COVID X-ray Modified," *Kaggle*, 2021. https://www.kaggle.com/datasets/suddirutten/covid-xray-modified (accessed Nov. 18, 2021).

[23] A. Musha, "COVID19 Customized X-ray Dataset," *Kaggle*, 2021. https://www.kaggle.com/datasets/mushaxyz/covid19-customized-xray-dataset (accessed Nov. 18, 2021).

[24] S. Sen, "Novel COVID-19 Chestxray Repository," *Kaggle*, 2021. https://www.kaggle.com/datasets/subhankarsen/novel-covid19-chestxray-repository (accessed Nov. 18, 2021).

[25] S. Gupta, "COVID-19 X-Ray Dataset With Preprocessed Images," *Kaggle*, 2020. https://www.kaggle.com/datasets/shreyanshgupta/covid19-xray-dataset-with-preprocessed-images (accessed Nov. 18, 2021).

[26] E. Vantaggiato, "Covid-19 X-ray - Two proposed Databases," *Kaggle*, 2021. https://www.kaggle.com/datasets/edoardovantaggiato/covid19-xray-two-proposed-databases (accessed Nov. 18, 2021).

[27] Jtiptj, "Chest X-Ray (Pneumonia,Covid-19,Tuberculosis)," *Kaggle*, 2021. https://www.kaggle.com/datasets/jtiptj/chest-xray-pneumoniacovid19tuberculosis (accessed Nov. 18, 2021).