



Pneumonia Image Classification Using CNN with Max Pooling and Average Pooling

¹Annisa Fitria Nurjannah, ²Andi Shafira Dyah Kurniasari, ³Zamah Sari, ⁴Yufis Azhar

^{1,2,3,4}Informatics, Faculty of Engineering, University of Muhammadiyah Malang

¹annisafitria2102@gmail.com, ²andishafira757@gmail.com, ³zamahsari@umm.ac.id, ⁴yufis@umm.ac.id

Abstract

Pneumonia is still a frequent cause of death in hundreds of thousands of children in most developing countries and generally detected clinically through chest radiographs. This method still difficult to detect the disease and requires a long time to produce a diagnosis. To simplify and shorten the detection process, we need a faster method and more precise in diagnosing pneumonia. This study aims to classify chest x-ray images using the CNN method to diagnose pneumonia. The proposed CNN model will be tested using max & average pooling. The proposed model is a development of the model in previous studies by adding batch normalization, dropout layer, and the number of epochs used. To maximize model performance, the dataset used will be optimized with oversampling & data augmentation techniques. The dataset used in this study is "Chest X-Ray Images (Pneumonia)" with a total of 5,856 data divided into two classes, namely Normal and Pneumonia. The proposed model gets 98% results using average pooling where the results increase by 9-13% better than the previous study. This is because overall pixel value of the image is highly considered to classify normal lungs and pneumonia.

Keywords: pneumonia, image classification, Convolutional Neural Network

1. Introduction

Pneumonia is an inflammatory lung disease caused by bacteria. A person suffering from this disease will have symptoms in the form of a very fast breathing frequency, headache, high fever accompanied by coughing up phlegm, shortness of breath, restlessness, and lack of appetite [1]. Not only that, but pneumonia also occurs due to viral and fungal infections. According to UNICEF records in 2018, this disease is the leading cause of under-five mortality in the world [2]. Supported by Indonesia's 2019 health profile, pneumonia is still the main problem causing 979 deaths, followed by diarrheal disease afterward. Then, the World Health Organization (WHO) stated that in 2017, as many as 800,000 children died from pneumonia [3].

To confirm/diagnose the presence of pneumonia in a person's body, the doctor will perform a series of examinations, including chest radiographs, CT-Scans, and Magnetic Resonance Imaging (MRI) [4]. The examination method with chest radiograph is one of the most frequently used examinations in the form of medical images because it is more affordable [5]. The reading of the image from the chest radiograph has a drawback in the form of difficulty in detecting the

disease so it takes a long time before medical personnel or doctors can diagnose the disease suffered by the patient.

The World Health Organization (WHO) describes that health information systems can assist the decision-making process to detect and control health problems [6], where this has a link to the disease prevention process based on existing risk factors. To control the risk of pneumonia and determine the presence of the disease in each individual, a classification process is needed. Medical images play a very important role in the process of classifying and identifying a disease [7]. The classification technique that can be implemented on chest radiograph image data is a classification technique using machine learning technology. Machine learning is used to train machines to handle big data more efficiently, where the goal of machine learning is to learn data so that it can be developed to learn without any direction from the user [8]. One part of this technology is deep learning, where the technique can be used quickly and efficiently to diagnose various diseases with a fairly good level of accuracy [9].

Convolutional Neural Network (CNN) is included in the type of deep learning because it has a deep neural

network. CNN has an algorithm that is used to get a new representative information from the convolution results, where the matching is done on each part of the image and then extracted [10]. The CNN structure consists of input, feature extraction, classification, and output. The extraction process in CNN consists of several hidden layers, namely the convolution layer, activation function (ReLU), and pooling. CNN works in layers so that the results in the first convolution layer will be used as input for the next convolution layer [11]. The classification process consists of fully-connected and activation functions (sigmoid) which has an output in the form of classification results [12].

In a previous study, CNN with augmented data was implemented to detect pneumonia based on a person's Chest X-Ray image and resulted in an accuracy of 85% [13]. In addition, the training process carried out in this study only used 10 epoch values and produced a CNN model that experienced both overfitting and underfitting at the same time. Then, the dataset used in this study is still raw data that has not been processed so it is still in an unbalanced condition.

One way to handle the conditions mentioned above is to use the pooling layer in the CNN structure. The purpose of using the pooling layer is to reduce the dimensions of the feature map (down sampling) so that it can speed up computing considering that the parameters used are getting less and less to update [14]. There are several types of pooling layers that are commonly used, such as max pooling, average pooling, global max pooling, global average pooling, and many more. However, previous studies have stated that the combination of max pooling and average pooling can increase generalizability of the CNN architecture [15]. In addition, the types of max and average pooling are the most widely used and implemented pooling layers in the world of computer vision [16].

In addition to the pooling layer, previous studies have stated that problems related to the number of unbalanced datasets can be overcome using oversampling techniques by randomly duplicating datasets in fewer classes than other classes [17]. The data taken on the minority class is the difference between the minority class and the majority class. After this is done, the dataset in each class will be balanced.

Based on the facts mentioned and previous research, this study will optimize the model built on the Convolutional Neural Network with data augmentation techniques and handling unbalanced dataset conditions using oversampling techniques. Data augmentation is intended to manipulate images and change the model architecture, especially the type of pooling layer in the form of max pooling and average pooling in the two same model architectures. Max pooling will divide the feature map into several sub-sections and reduce the

part in new sub-sections and choose the maximum value from the feature map. The same applies to the selection of average pooling, but this type of layer will choose the average value of the feature map to be able to retrieve all image information. Then, this study will share train, validation, and test data, utilize the dropout layer, and increase the number of epochs in the training dataset process in classifying someone with or without pneumonia.

2. Research Methods

2.1 Research Stages

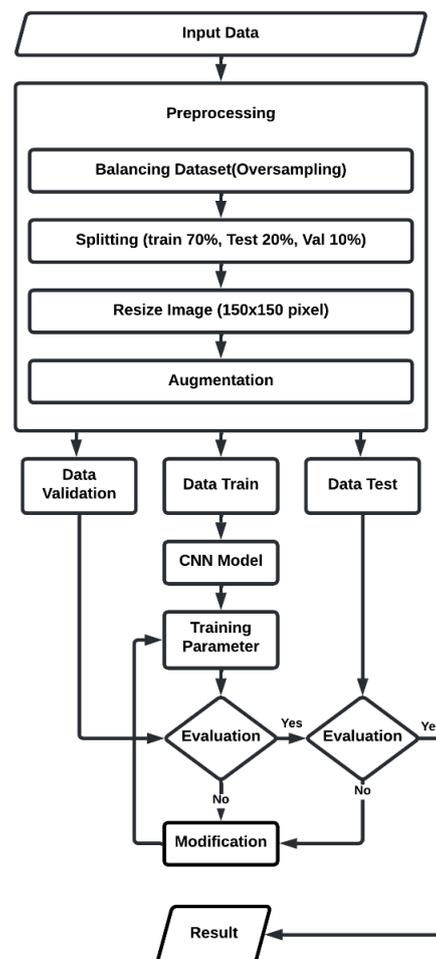


Figure 1. Research stages

The stages of the research carried out are listed in Figure 1. The research starts by getting the dataset which is then preprocessed which is divided into several other sub-processes. At this stage, the dataset used is still an imbalanced dataset, so an oversampling technique will be used to overcome this problem. After that, the next preprocessing stage is data splitting, which is dividing the "Chest X-Ray Images (Pneumonia) dataset into three folders, namely data train, validation, and test. Then, the next sub-process is resizing the dataset, which is modifying the size of the entire image in the dataset

from $1,500 \times 1,500$ pixels to 150×150 pixels. The last stage of preprocessing is to perform dataset augmentation.

The next stage is to process the results of data grouping after preprocessing. Train data is needed to train data with the model and parameters that have been built, while data validation is used to validate the training results with the parameters that have been determined in the model so that if there are underfitting or overfitting conditions, the model and parameters can be evaluated before being tested on the test data.

2.2 Dataset

In this study, the dataset used is a dataset available on the Kaggle page with the title "Chest X-Ray Images (Pneumonia)" [18]. Based on Figures 2 and 3, there are two categories of images, namely Normal and Pneumonia, where the entire dataset is 5,856 data with a size of $1,500 \times 1,500$ pixels.

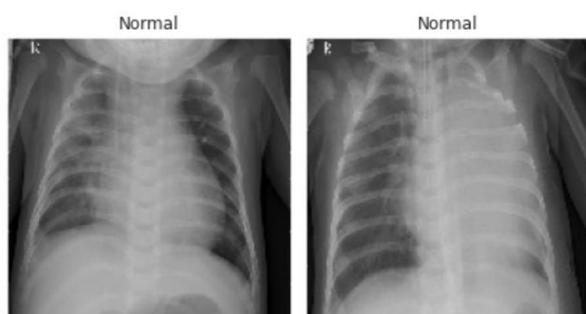


Figure 2. Normal chest X-Ray sample

Figure 2 is a sample image with a normal chest category totaling 1,583 data, while Figure 3 is a sample image with a chest category with pneumonia totaling 4,273 data. Each image has a JPEG format.

The dataset is still not balanced so an oversampling technique will be used to deal with this problem. The oversampling technique used is to add data in fewer classes as much as the difference between the amount of data in that class and the data in the largest class.

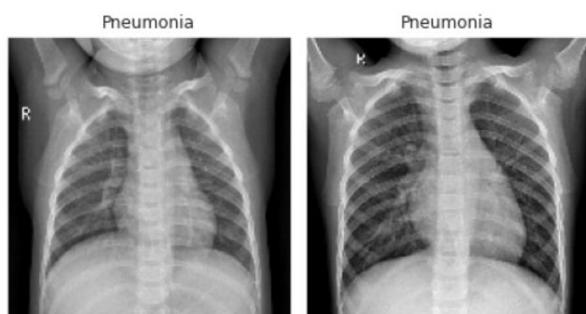


Figure 3. Samples of chest X-ray results with pneumonia

2.3. Model Architecture

The researcher proposes two handcraft models as a comparative test to be carried out. The first model is a CNN model which has an input layer on the convolution

layer with a size of 150×150 pixels and 1 channel. The pixel size of the image in the input layer, which was originally 1500×1500 pixels, was changed to 150×150 pixels, aimed at making the image more focused on objects that act as markers of pneumonia or not. While the selection of channel size is based on the type of image in the form of a grayscale image.

The architecture of model 1 can be seen in Figure 4, where the second layer in the first model is the Batch Normalization layer which is used to speed up and stabilize the training dataset process [19]. The coating was used five times. Furthermore, the model implements the pooling layer five times where in this pooling layer two types of tests will be carried out using max pooling and average pooling which have a filter of 2×2 , the kernel displacement distance of 2. convolution five times with each filter size of 32, 64, 64, 128, and 256, as well as the type of relu activation. Then, the fully connected layer is composed of a flatten layer, a dense layer with a random number of 128 and a relu activation type, and a dropout layer with a probability value of 0.2 which aims to handle the possibility of overfitting conditions. Finally, there is an output layer with a random number of 1 which is intended for grayscale images and sigmoid activation types, considering the number of classes used is 2.

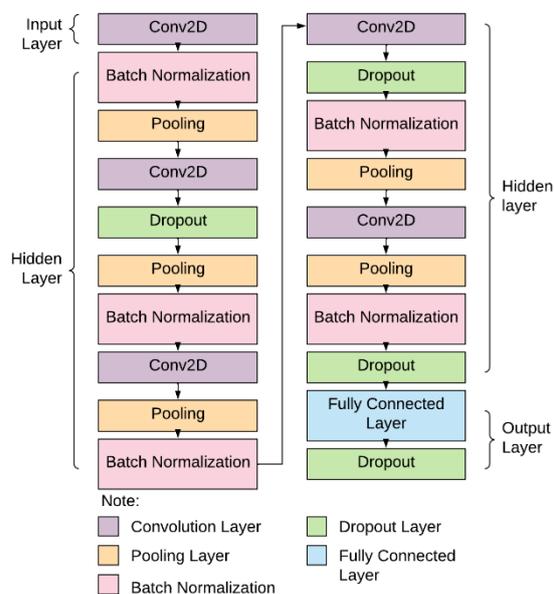


Figure 4. First model architecture

The second model is also a handcrafted CNN model which has an input layer on the convolution layer measuring 150×150 pixels with 1 channel. The architecture of model 2 can be seen in Figure 5. The second layer is a dropout layer which is used to handle overfitting. The dropout layer in model two is used three times, each of which has a probability value of 0.2. In addition, there is also a pooling layer used four times in model 2. Similar to model 1, the second model will be

tested using two types of pooling, namely max pooling and average pooling. Furthermore, model 2 applies three convolution layers with each filter, namely 16, 32, and 64 with the type of relu activation. Then, the fully connected layer is composed of a flatten layer, a dense layer with a random number of 128, and the type of relu activation. Finally, there is an output layer with a random number of 1 which is intended for grayscale images and sigmoid activation types, considering the number of classes used is 2. The architecture of model 2 can be seen in Figure 5.

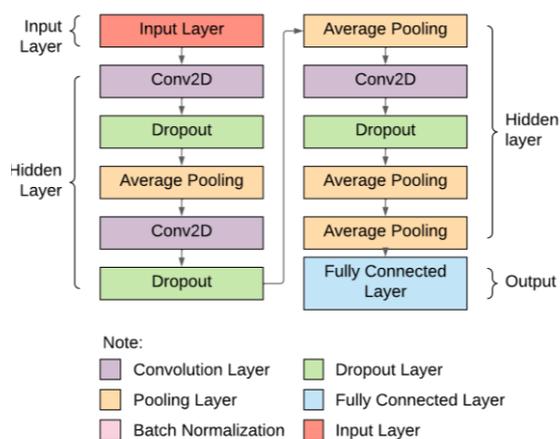


Figure 5. Second model architecture

2.4 Data Augmentation

All images in the dataset will be partially augmented at random with the aim that the system can still recognize various forms of images as data for patients with normal chest and pneumonia when classification will be carried out.

The augmentation process uses the Image Data Generator function with several parameters activated following Table 1 which will change the shape of each image in the dataset.

Table 1. Types of Parameters Activated in the Augmentation Process

Parameter	Server
feature_center	False
samplewise_center	False
featurewise_std_normalization	False
samplewise_std_normalization	False
zca_whitening	False
rotation_range	30°
zoom_range	0,2
width_shift_range	0,1
height_shift_range	0,1
horizontal_flip	True
vertical_flip	False

2.5 Test Scenario

In this study, two classes will be classified, namely individuals with and without pneumonia (Normal/Pneumonia). Furthermore, the dataset will be duplicated randomly (oversampling) and divided into

three types of data in the form of train, validation, and test data which will be used for training and model testing.

The total distribution of data is summarized in Tables 2 and 3 with the distribution of train, validation, and test data respectively at 70%, 10%, and 20% of the total data in each class. The number of datasets before and after the augmentation process is the same, this is because the augmentation process does not add image variations, but manipulates each existing image into various forms. Table 2 is the amount of train, validation, and test data before augmentation & oversampling is performed.

Table 2. Amount of Data Before Oversampling & Data

Class	Augmentation Process			Total Images
	Training Images	Validation Images	Testing Images	
Normal	1.102	157	316	1.583
Pneumonia	2.986	426	854	4.273

Then, Table 3 is the number of train, validation, and test data after the augmentation and oversampling processes are carried out at once.

Table 3. Amount of Data After Oversampling & Data Augmentation Process

Class	Training Images	Validation Images	Testing Images	Total of Images
Normal	2.991	427	855	4.273
Pneumonia	2.991	427	855	4.273

Furthermore, each model is tested in two scenarios, each of which applies a different type of pooling, namely max pooling and average pooling according to Table 4.

Table 4. Test Scenario

Model	Total of Hidden Layer	Type of Pooling Layer in First Scenario	Type of Pooling Layer in Second Scenario
Model 1	4	Max Pooling	Average Pooling
Model 2	3	Max Pooling	Average Pooling

3. Results and Discussions

The results of this study are the stages carried out based on the arrangement of methods that have been described in the research method. Dataset selection is the first step in this research. The dataset with the title Chest X-Ray Images (Pneumonia) consists of two classes, namely Normal with 1,583 data and Pneumonia with 4,273 data. The graph of the amount of data for each class is summarized in Figure 6.

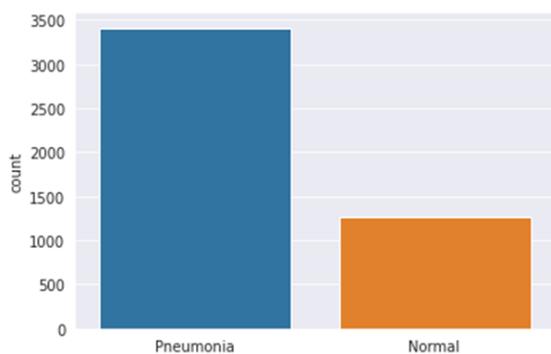


Figure 6. Graph of the amount of data for each class before oversampling

Next, the dataset is downloaded and saved into Google Drive using an API integrated with Kaggle via Google Collaboratory. Google Collaboratory was chosen because it has collaboration features so that researchers can easily share projects. Before processing, the dataset will be handled with an oversampling technique to balance the amount of data in each class.

The Normal class has less data than the Pneumonia class, so the amount of data in the Normal class will be added as much as the difference from the amount of data between the Normal and Pneumonia classes, which is 2,690 data. The graph of the amount of data after oversampling is summarized in Figure 7.

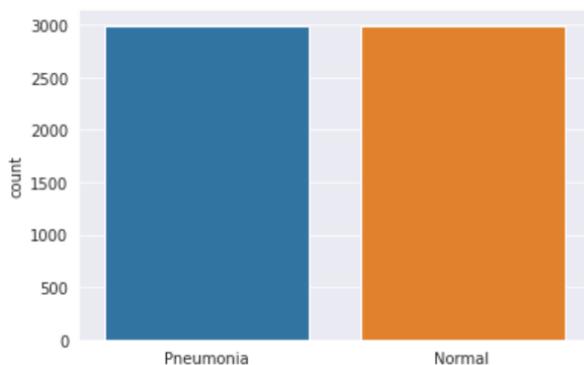


Figure 7. Graph of the amount of data for each class after oversampling

Furthermore, because the dataset used is still in .zip format, extraction is carried out on the entire dataset. Then, the datasets were redistributed to train, validation, and test data with a ratio of 70%, 10%, and 20% after oversampling.

The next process is to normalize the dataset used by dividing each image by 255, where that number is the maximum image intensity so that the brightness of the image is in the range of 0-1. Next, resize the data for each data in the train, validation, and test data. At this stage, the channel size of each image is changed to 1 so that each image has only one layer.

The next step is to perform several data augmentation techniques, namely randomly rotating the image by 30 degrees, zooming at random by 20% and shifting the

horizontal and vertical images randomly. Augmentation of data in this study aims to increase the variety of data images held on training data and validation data. An example of the results of data augmentation can be seen in Figure 8.

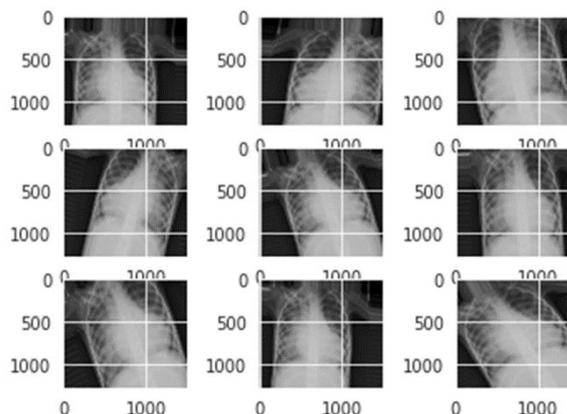


Figure 8. An example image of the augmentation result

Next, the datasets are labeled 0 and 1, where 0 is the type of normal chest X-ray and 1 is the type of chest X-ray with pneumonia.

3.1 Model Performance Comparison

In the next stage, each model that has been built is run using Adam as the optimizer to update the network weight iteratively based on the training data. Furthermore, the model is trained using the loss `binary_crossentropy` parameter, batch size of 32, a learning rate of 0.0001, and 100 epochs. After training the model, the two model scenarios give different accuracy results, where the results given are summarized in Table 5.

Table 5. Accuracy Results of Each Scenario

Model	Type of Pooling Layer	Accuracy
Model 1	Max Pooling	98%
Model 1	Average Pooling	98%
Model 2	Max Pooling	98%
Model 2	Average Pooling	98%

Of the four experiments carried out, all models with each implemented pooling have the same results when referring to Table 5 as an accuracy table. This means that both models with each pooling used to have a predictive value that is quite close to the actual value indicated by the accuracy value which only has a distance of 2% to 100%.

However, if re-analyzed, each experiment has different precision and recall value. Based on Table 6, the precision value in model 1 with average pooling has an increased value of 1% compared to model 1 with max pooling when predicting the Pneumonia class correctly among all actual classes. While the recall value in model 1 with average pooling also has the same

increase when predicting the Normal class among all the actual Normal classes.

Table 6. Precision Results and Recall of Each Scenario

Model	Class	Precision	Recall
Model 1 Max Pooling	Normal	98%	97%
	Pneumonia	97%	98%
Model 1 Average Pooling	Normal	98%	98%
	Pneumonia	98%	98%
Model 2 Max Pooling	Normal	98%	97%
	Pneumonia	97%	98%
Model 2 Average Pooling	Normal	99%	97%
	Pneumonia	97%	99%

This happens in model 2 with average pooling, where the precision value obtained in the Normal class increases by 1% compared to model 2 with max pooling when predicting the Normal class correctly among all actual classes. Then, the recall value in the Pneumonia class also increased by 1% when implementing model 2 with average pooling.

In the training process that has been carried out, model 1 with max pooling shows the graphs of loss and accuracy results listed in Figures 9 and 10. The resulting graphs of loss results and accuracy results are indicated to be overfit with a val_accuracy value of 84.43% and a val_loss that is still quite large so that the value of val_accuracy has not yet reached the range of 90-100%.

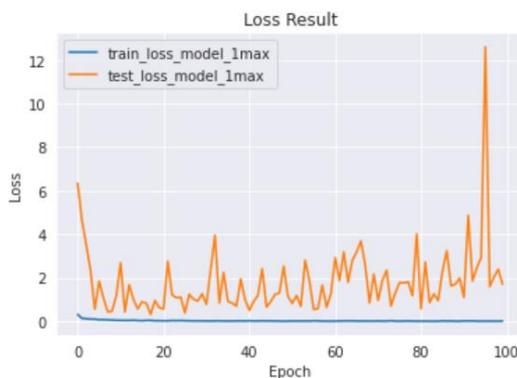


Figure 9. Loss result chart for model 1 with max pooling

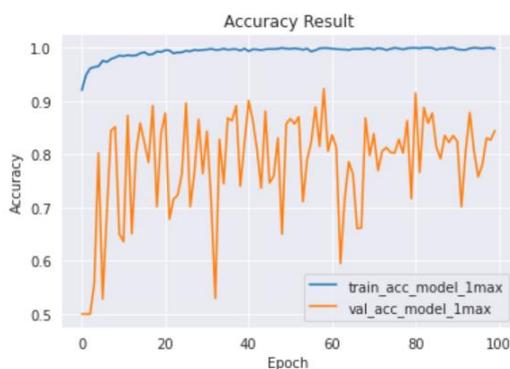


Figure 10. Graph of accuracy result model 1 with max pooling

Then, model 1 with average pooling in Figure 11 and Figure 12 each shows the loss result and accuracy result values which are still overfitted, but appear more stable. The val_accuracy value generated in this model is 88.41%. However, the resulting val_loss value becomes lower and the val_accuracy becomes higher than the model with the previous pooling type.

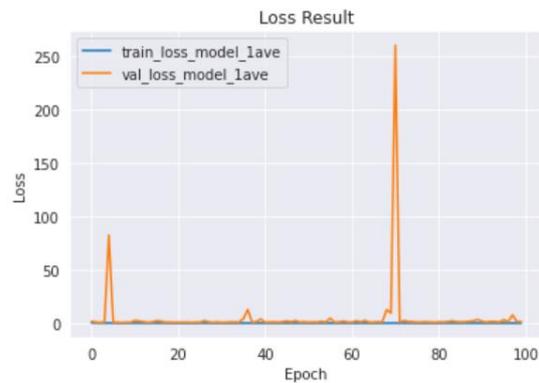


Figure 11. Loss result chart for model 1 with average pooling

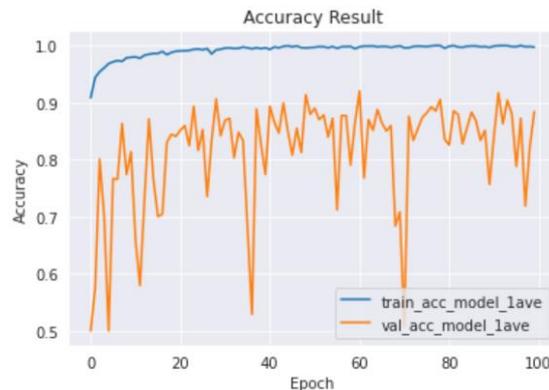


Figure 12. Graph of accuracy result model 1 with average pooling

Furthermore, Figure 13 and Figure 14 show the loss value and accuracy result of model 2 when using max pooling which is still in an overfitting condition. The resulting val_accuracy value is 77.28% with a val_loss value which is still relatively high.

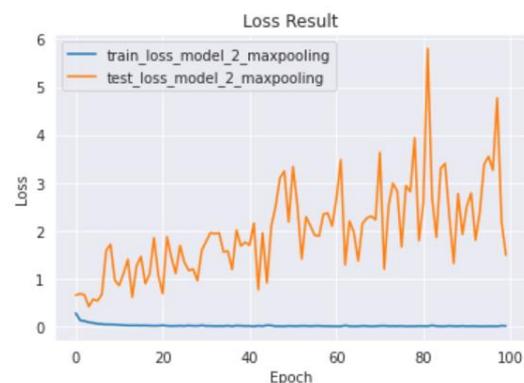


Figure 13. Loss result chart for model 2 with max pooling

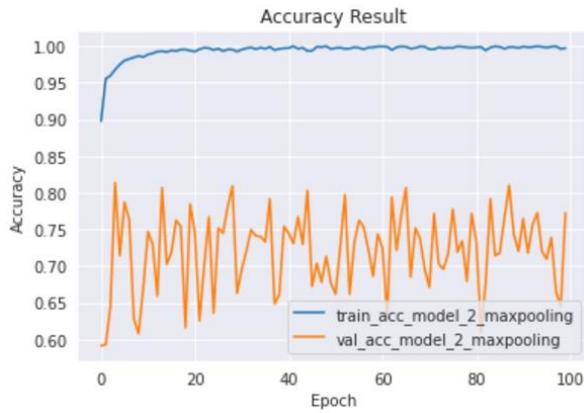


Figure 14. Accuracy result chart for model 2 with max pooling

While in Figure 15 and Figure 16, the graphs of loss result and accuracy result model 2 with average pooling are in the same condition with the val_accuracy value increasing by 0.7916.

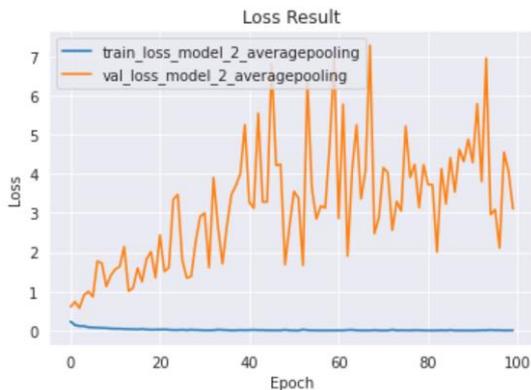


Figure 15. Loss result chart for model 2 with average pooling

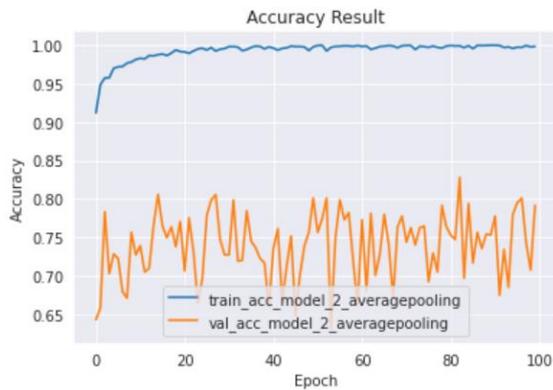


Figure 16. Accuracy result chart for model 2 with average pooling

After doing the 4 experiments above, the use of average pooling in the Convolutional Neural Network model layer is better than the use of max pooling. This can be proven by the decreasing value of val_loss in model 1 and val_accuracy which always increases in both models.

Then, there are several loss values that experience sudden spikes in model 1 with each type of pooling. This condition can be caused by the model always

changing in every running epoch. The model will always change during the training process in the epoch because it will always optimize each parameter used to increase accuracy so that the accuracy value can still change as the epoch goes. In addition, the spike can be caused by the batch_size value that does not match the number of datasets used.

Furthermore, the four experiments still tend to experience overfitting. This can be caused by several possibilities, such as the model being compiled is still too complex with many layers, the batch normalization parameters are not correct, the dropout layer is located among other layers, the amount of training data, and the number of epochs used.

3.2 Confusion Matrix Results

The confusion matrix was created to measure the performance of the classification method that has been built. Figure 17 is a confusion matrix table generated from model 1. Based on the table, it can be understood that in the Pneumonia class there are 825 image data that were predicted correctly and 29 image data that were predicted incorrectly. And in the Normal class there are 303 data that were predicted correctly and 13 data that were incorrectly predicted by the model. Then, Figure 18 is a table of confusion matrix results from model 2.

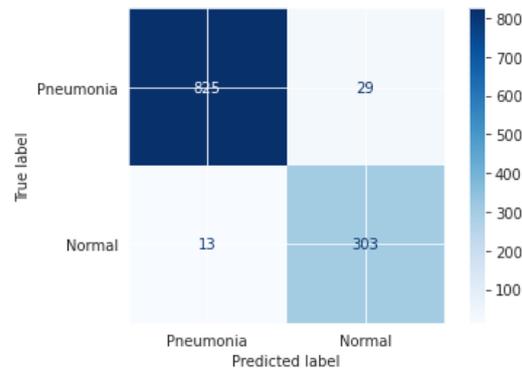


Figure 17. Confusion matrix for model 1

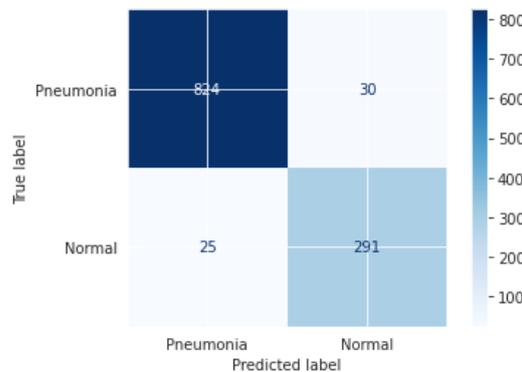


Figure 18. Confusion matrix for model 2

Based on the confusion matrix model 2 table, it can be concluded that in the Pneumonia class there are 824

image data that are predicted correctly and 30 image data that is predicted incorrectly. And the Normal class there are 291 image data predicted correctly and 25 images predicted incorrectly by the model.

3.3. Comparison of the Best Model Performance with Previous Research

After carrying out a series of test scenarios, the next process is to compare the performance of the best model with the results obtained in previous studies. Based on Figure 19, the classification report in model 2 when using average pooling obtains an accuracy of 98% and a precision of 99% in the Normal class and 97% in the Pneumonia class, where the data increases overall after using the model with max pooling.

	precision	recall	f1-score	support
0	0.99	0.97	0.98	855
1	0.97	0.99	0.98	855
accuracy			0.98	1710
macro avg	0.98	0.98	0.98	1710
weighted avg	0.98	0.98	0.98	1710

Figure 19. Classification report model 2 with average pooling

According to Table 7, this study produces a model in scenario 2 as the best model, where this model can exceed the accuracy of the model built in previous studies by 9-13%.

Table 7. Accuracy Results of Each Scenario

Model	Accuracy
CNN with handcrafted model [13]	85%
CNN with ResNet15 [17]	89%
The best proposed model	98%

3.4. Testing

Model 2 with average pooling being the best model is then retested using several datasets. In Figure 20, the results given show that as many as 5 image data can be predicted in the Pneumonia class correctly and achieve an average accuracy of 100%. in the average prediction time of 0.0586 seconds.

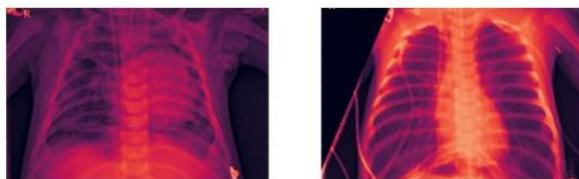


Figure 20. Sample images from testing results in the Pneumonia class

Then, Figure 21 shows that the next 5 data were successfully predicted in the Normal class correctly and achieved an average accuracy of 99.99% in an average prediction time of 0.0544 seconds.

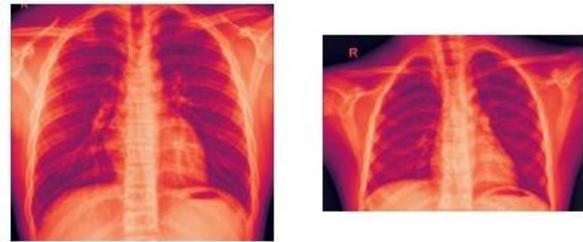


Figure 21. Sample images from testing results in the Normal class

4. Conclusion

Based on the research that has been carried out when implementing two models and two scenarios, it shows that both models with average pooling and max pooling scenarios get an accuracy of 98%, where the results are better than previous studies which have an accuracy value of 89% [17] and 85% [13]. In both scenarios, average pooling and max pooling get the same accuracy results in both models. However, when viewed from the precision and recall values produced, model 2 with average pooling gets better results. Therefore, the application of average pooling tends to be better applied to the classification of CT-scan Pneumonia images, this is because during model training, information from the entire image is considered to classify normal lungs and pneumonia.

Then, to continue this research, it is recommended that future studies reduce the potential for overfit conditions by using a callback or learning rate scheduler. This is intended to find out the best number of epochs when the val_accuracy value is no longer increasing or the val_loss value is no longer decreasing after a certain number of epochs.

Acknowledgment

Thank you to all those who have helped during the writing process of this journal. Thanks are conveyed to; Allah SWT who has launched the entire process of writing the journal, Mr. Zamah Sari as the first supervisor, Mr. Yufis Azhar as the supervisor for the machine learning course and the second supervisor, as well as all friends who always support and encourage during the writing process until the end of the journal writing this.

Reference

- [1] R. Irawan, R. Reviono, and H. Harsini, "Correlation Between Copeptin and PSI with Intravenous to Oral Antibiotic Switch Therapy and Length of Stay in Community-Acquired Pneumonia," *J. Respirologi Indones.*, vol. 39, no. 1, pp. 44–53, Jan. 2020, doi: 10.36497/jri.v39i1.40.
- [2] R. Villar and S. Sidhu, "Lembaga kesehatan dan anak memperingatkan satu anak meninggal akibat pneumonia setiap 39 detik," *UNICEF Indonesia*, 2019. <https://www.unicef.org/indonesia/id/press-releases/lembaga-kesehatan-dan-anak-memperingatkan-satu-anak-meninggal-akibat-pneumonia-setiap-39-detik#:~:text=Pada tahun 2018%2C angka kematian,malaria merenggut nyawa 272.000 anak.> (accessed

- Jun. 18, 2021).
- [3] Biro Komunikasi dan Pelayanan Masyarakat, "Pneumonia Pada Anak bisa Dicegah dan Diobati," *Kementrian Kesehatan Republik Indonesia*, 2020. <https://www.kemkes.go.id/article/view/20111500001/pneumonia-pada-anak-bisa-dicegah-dan-diobati.html> (accessed Jan. 14, 2022).
- [4] M. Dame Cristy Pane, "Pneumonia," *Alodokter*, 2020. <https://www.alodokter.com/pneumonia/diagnosis> (accessed Jan. 17, 2022).
- [5] P. A. H. Pratama, R. Teguh, A. S. Sahay, and V. Wilentine, "Deteksi COVID-19 Berdasarkan Hasil Rontgen Dada (Chest Xray) Menggunakan Python," *J. Inf. Technol. Comput. Sci.*, vol. 1, no. 1, pp. 58–67, 2021, doi: 10.47111/jointecom.v1i1.2956.
- [6] Kemenkes RI, "Stroke Dont Be The One." p. 10, 2018.
- [7] I. M. D. Maysanjaya, "Klasifikasi Pneumonia pada Citra X-rays Paru-paru dengan Convolutional Neural Network," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 2, pp. 190–195, 2020, doi: 10.22146/jnteti.v9i2.66.
- [8] M. Batta, "Machine Learning Algorithms - A Review," *Int. J. Sci. Res. (IJ)*, vol. 9, no. 1, pp. 381–undefined, 2020, doi: 10.21275/ART20203995.
- [9] D. Singh, V. Kumar, Vaishali, and M. Kaur, "Classification of COVID-19 patients from chest CT images using multi-objective differential evolution-based convolutional neural networks," *Eur. J. Clin. Microbiol. Infect. Dis.*, vol. 39, no. 7, pp. 1379–1389, 2020, doi: 10.1007/s10096-020-03901-z.
- [10] M. R. Fauzi, P. Eosina, and D. Primasari, "Deteksi Coronavirus Disease Pada X-Ray Dan CT-Scan Paru Menggunakan Convolutional Neural Network," *JUSS (Jurnal Sains dan Sist. Informasi)*, vol. 3, no. 2, pp. 17–27, Mar. 2021, doi: 10.22437/juss.v3i2.10888.
- [11] M. H. Romario, E. Ihsanto, and T. M. Kadarina, "Sistem Hitung dan Klasifikasi Objek dengan Metode Convolutional Neural Network," *J. Teknol. Elektro*, vol. 11, no. 2, p. 108, 2020, doi: 10.22441/jte.2020.v11i2.007.
- [12] A. Peryanto, A. Yudhana, and R. Umar, "Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation," *J. Appl. Informatics Comput.*, vol. 4, no. 1, pp. 45–51, 2020, doi: 10.30871/jaic.v4i1.2017.
- [13] J. Garstka and M. Strzelecki, "Pneumonia detection in X-ray chest images based on convolutional neural networks and data augmentation methods," *Signal Process. - Algorithms, Archit. Arrange. Appl. Conf. Proceedings, SPA*, vol. 2020-Septe, pp. 18–23, 2020, doi: 10.23919/spa50552.2020.9241305.
- [14] M. R. Alwanda, R. P. K. Ramadhan, and D. Alamsyah, "Implementasi Metode Convolutional Neural Network Menggunakan Arsitektur LeNet-5 untuk Pengenalan Doodle," *J. Algoritm.*, vol. 1, no. 1, pp. 45–56, 2020, doi: 10.35957/algorithm.v1i1.434.
- [15] M. Momeny, A. Jahanbakhshi, K. Jafarnejhad, and Y. D. Zhang, "Accurate classification of cherry fruit using deep CNN based on hybrid pooling approach," *Postharvest Biol. Technol.*, vol. 166, no. April, p. 111204, 2020, doi: 10.1016/j.postharvbio.2020.111204.
- [16] E. N. Arrofiqoh and H. Harintaka, "Implementasi Metode Convolutional Neural Network Untuk Klasifikasi Tanaman Pada Citra Resolusi Tinggi," *Geomatika*, vol. 24, no. 2, p. 61, 2018, doi: 10.24895/jig.2018.24-2.810.
- [17] A. H. Ahnafi, A. Arifianto, and K. N. Ramadhani, "Pneumonia Classification from X-ray Images using Residual Neural Network," *Ind. J. Comput.*, vol. 5, no. September, pp. 43–54, 2020, doi: 10.21108/indoic.2020.5.2.454.
- [18] P. Mooney, "Chest X-Ray Images (Pneumonia)," *Kaggle*, 2018. <https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia> (accessed Mar. 26, 2021).
- [19] J. Huber, "Batch normalization in 3 levels of understanding," *Towards Data Science*, 2020. <https://towardsdatascience.com/batch-normalization-in-3-levels-of-understanding-14c2da90a338> (accessed Jan. 25, 2022).