International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 16, No.1 (Oct-2024)

http://dx.doi.org/10.12785/ijcds/1601124

Pneumonia Medical Image Classification Using Convolution Neural Network Model AlexNet & GoogleNet

Rio Subandi[1], Herman[2] and Anton Yudhana[3]

[1,2]*Master of Informatics Engineering Study Program Ahmad Dahlan University, Yogyakarta, Indonesia*
[3]*Department of Electrical Engineering University of Ahmad Dahlan, Yogyakarta, Indonesia*

Received 24 Feb. 2024, Revised 28 Jun. 2024, Accepted 2 Aug. 2024, Published 1 Oct. 2024

Abstract: Pneumonia is one of the deadliest diseases in the world. Diagnosis of pneumonia is done with the help of CT-scan image analysis of the chest. This analysis is usually done by a pulmonary specialist. The availability of pulmonary specialists is still limited, especially in underdeveloped, outermost and frontier areas. In addition, manual analysis still faces the possibility of errors. The use of artificial intelligence technology is expected to overcome these problems. The purpose of this study is to obtain the results of pneumonia disease classification using the CNN algorithm using the AlexNet and GoogleNet models. The tools used in this research are python. The image dataset used amounted to 5856 images obtained from the Kaggle repository. The stages of this research consist of data preparation where this data has been preprocessed and split data. Furthermore, the CNN stage with the architecture used is AlexNet and GoogleNet. The training data used is 90% of the data or 5270 images and the testing data is 10% or 586 images. The training model is done in as many as 20 iterations so that the model used can recognize the image more accurately. After the model has been trained the model will be tested by providing test data. The results of this research are displayed in the confusion matrix. The results of the research using the AlexNet and GoogleNet architectures get an accuracy value. This accuracy value is then compared between the two. The accuracy obtained from AlexNet architecture is 96% while that obtained from GoogleNet is 94%. From the results of the accuracy of the two models, it can be concluded that the AlexNet architecture has the highest accuracy of 96%.

Keywords: AlexNet, GoogleNet, Pneumonia, Classification

1. INTRODUCTION

Pneumonia can pose a significant danger to health [1]. The process of diagnosing pneumonia involves various steps taken by the doctor to identify the infection in the lungs [2]. First, the doctor will gather information about the patient's symptoms and medical history. Next, a physical examination is performed to look for signs of infection in the lungs, such as abnormal breathing sounds. Radiological examinations, such as chest X-rays or CT scans, are used to see a clearer picture of the lungs. Doctors may also examine the patient's sputum samples to determine the cause of the infection, as well as conduct blood tests to measure the level of white blood cells that could indicate an infection. Once a diagnosis of pneumonia is established, appropriate treatment can be recommended by the doctor according to the cause and severity of the infection [3]. Prevention of pneumonia is also important, such as getting vaccinated and maintaining hand hygiene to reduce the risk of developing the disease. Early diagnosis and proper treatment are essential to effectively treat pneumonia. Doctors may perform a physical examination, blood analysis, chest X-ray, or other tests to identify a lung infection. Treatment usually involves antibiotics for bacterial pneumonia and symptomatic treatments to relieve symptoms [1].

Convolutional Neural Network (CNN) is a type of deep learning algorithm that has brought revolutionary changes in the world of healthcare, especially in diagnosing pneumonia [1]. The use of CNNs in the medical field, particularly for the analysis of medical images such as chest X-rays, has enabled early and accurate detection of this lung disease. One of the challenges in diagnosing pneumonia from X-ray images is the complexity of the structures and patterns that doctors must identify. CNNs can overcome this problem by repeatedly training using chest X-ray data that has been correctly classified by an expert. CNN will automatically extract important features from X-ray images to understand the important characteristics that indicate the presence of pneumonia infection. The trained CNN algorithm can analyze new chest X-ray images and accurately identify signs of pneumonia [3]. This allows for earlier detection and more timely intervention, thus speeding up the diagnosis and treatment of patients. In addition, the use of CNNs can help reduce the potential for human error in image interpretation, optimize the reliability of diagnosis, and reduce the risk of errors in medical treatment. The positive influence of

E-mail address: riosubandi9@gmail.com, hermankaha@mti.uad.ac.id, eyudhana@ee.uad.ac.id

CNN in diagnosing pneumonia means improved diagnostic efficiency and accuracy. Faster and more precise results from these algorithms can enable early treatment and reduce complications and mortality caused by pneumonia. This will contribute to research and development in the field of artificial intelligence for medical diagnosis, particularly in medical image classification. Thus, the application of CNN in diagnosing pneumonia disease has opened up great opportunities in improving the quality of health care and reducing the burden of the disease in society [4].

In this research, the CNN model uses two architectures, namely GoogleNet and AlexNet. GoogleNet has 20 layers if we count only parameterized layers (or 27 layers if we also count pooling) [5]. This network in its native environment has been trained on over a million images and can classify images into 1000 object categories. The network architecture has learned a rich feature representation for various images. The network takes an image as input and outputs labels for objects in the image along with probabilities for each object category [6]. Eight layers make up the AlexNet architecture: three fully-connected layers and five convolutional layers, some of which are followed by max-pooling layers. With this network architecture, training performance is better than with ground and sigmoid networks because of the non-saturated ReLU activation function [7].

2. LITERATURE REVIEW

There are several previous studies related to the objects and methods in this study. Yopento et al [3] used the Sobel feature extraction-based Convolutional Neural Network approach to identify pneumonia in lung X-ray images. Lung image dataset objects are used in this study. The results were 91.54% accuracy, 91.8% recall, and 91% precision. With an epoch value of 50, a learning rate of 0.0001, and a batch value of 20, this study's accuracy rate was 91.54%. Abdillah et al [8] Using support vector machines and convolutional neural networks to classify viral pneumonia. The subjects of this study are two different types of images: lungs affected by bacterial pneumonia and lungs affected by viral pneumonia. The test results have an average accuracy of 0.85 according to the confusion matrix, so it can be concluded that the accuracy method is fairly high. The architecture and object methods are where this research differs from earlier studies. The CNN method and the subject of study are used in both prior research studies. Nurkhasanah Murinto [9] to classify facial skin diseases using the Convolutional Neural Network method. Objects used in facial image research. The results of the training process are 98% validation results with 325 image training data and 125 validation data. The accuracy result obtained when testing new data is 90% with 50 frames of test data. So it can be said that the results obtained in this research experiment are very good. Gatc Maspiyanti [10] Convolutional Neural Networks for the Prediction of Plasmodium Parasites in Microscopic Images of Red Blood Cells. The red blood cell image is the subject of this study. The test findings demonstrate good accuracy; specifically, the CNN algorithm model yields an accuracy score of 97.96 with a loss of 0.06 at roughly 121 seconds of computation time on average per epoch. Gong Kan [11] kidney tumors are divided and categorized using CNN. A 99.5% accuracy rate in the classification of benign and malignant tumors can be attained by combining 2D SCNet segmentation and classification. Dice coefficients of 0.946 and 0.846 were obtained in the three-label SCNet 2D segmentation results, respectively, suggesting that segmentation network learning benefits from the inclusion of a classification module.

Similarities from previous research both use the CNN method and some use the same object, namely pneumonia.

3. RESEARCH METHODS

The tools and software used to support the research can be seen in Table 1.

TABLE I. Tools and Software

No	Tools & Software	Versi	Usability
1	Laptop ROG Strix G513IM	-	Workstation to complete the research conducted
2	AMD Ryzen 7 4800 with Radeon	-	To perform turbo boost
3	Mouse and Keyboard	-	Tools for typing and hovering
4	Windows 11	64 bit	The workstation operating system
5	Python Jupyter Notebook 64 bit	3.10	Software tools
6	Web Browser 64 bit	106.0.1370.47	Software tools

The dataset used as material in the research is shown in Table 2.

TABLE II. Datasets

No	Material	Description
1	Dataset	X-ray image of pneumonia data
2	Amount of data	5856
3	Class	Normal and pneumonia
4	Format	JPEG
5	Size	185 x 1317, 2111 x 1509 and others

The flow of research conducted from start to finish is shown in Figure 1.

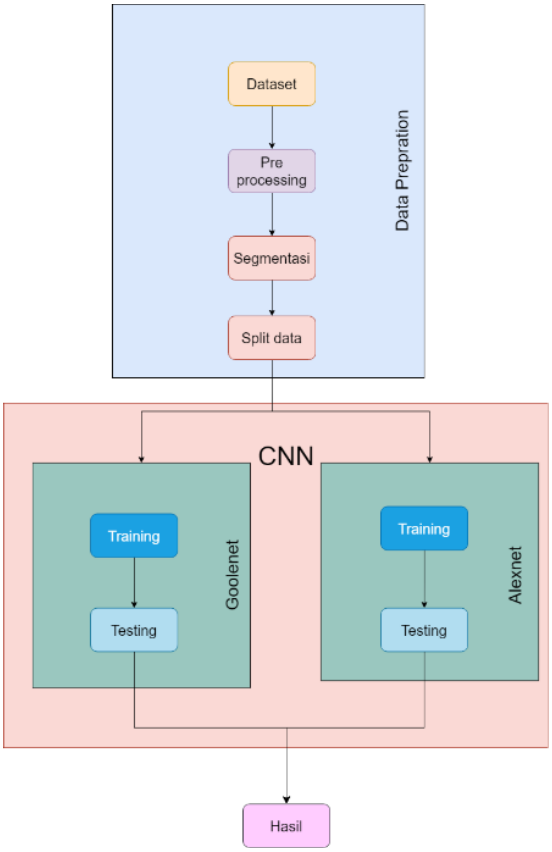

Figure 1. Research Flow

The research flow shown in Figure 1 shows the steps of the research flow in the initial stage, namely data preparation where at this stage the dataset will be preprocessed. preprocessing is done to transform the input image into a better image. Split data where the data is divided into 90% training data and 10% testing data. Segmentation divides the image into certain regions. After this data preparation stage, enter the CNN stage. This CNN uses two architectures, namely AlexNet and GoogleNet.

A. Datasets

This study uses a dataset of CT-Scan-generated pneumonia images. The dataset is used to train machine learning algorithms, such as Convolutional Neural Networks (CNN), in performing pattern recognition and classification of these images. The algorithms can learn important features of pneumonia CT-Scan images and accurately identify the location and size of the infection by utilizing deep learning technology. In addition, this dataset can also be used to develop decision support systems in the field of radiology, assisting doctors in diagnosing and planning patient treatment more precisely.

The use of pneumonia CT-Scan image datasets must consider the security and privacy aspects of the data. Since this medical data contains personal information of the patient, strict measures must be taken to protect the confidentiality and integrity of the data. Processing the dataset also requires trained radiologists to ensure the accuracy and reliability of the image annotations and diagnoses recorded in the dataset. Although pneumonia CT-Scan image datasets carry great potential in research and disease management, ethics and data security must remain a top priority to maintain trust and responsibility in their application. Based on this, this study uses a dataset from Kaggle that is freely used in research and has been labeled with images that have pneumonia and no disease. The dataset obtained is 5,856 which consists of two classes, namely normal and abnormal, with different image sizes 2090 x 2858, 1448 x 1056, and many other sizes. The format of the mind is jpeg, with a grayscale color model, 8-bit deep color. An example of the dataset used can be seen in Figure 2 (a)(b).

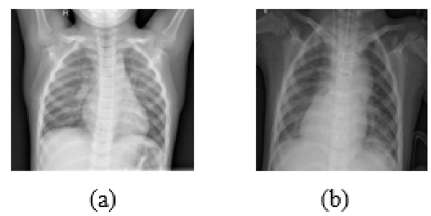

Figure 2. (a) Normal and (b) Pneumonia Thorax Rotgen Image

Figure 2 is a rotgen thorax image where this image will be used in this study. 2(a) is a normal image with an image size of 2090 x 2858, 2(b) is a pneumonia image with a size of 1448 x 1056.

B. Preprocessing

The datasets obtained have different sizes and types so data preprocessing is needed. The purpose of image preprocessing is to perform a series of steps and transformations on digital images before further processing [12]. Image preprocessing aims to improve image quality, remove irrelevant or distracting information, and optimize the image so that it is more suitable for subsequent analysis and processing. Preprocessing, several important things are done, including resizing and image conversion.

C. Resizing

Resizing is the process of changing the image size so that all images are the same size [13]. The image size used is 150x150. After all the data is resized to the same size, then the data is divided into 2, namely 90% training data and 10% testing data from the total dataset.

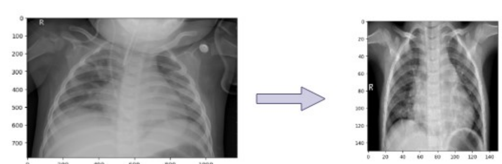

Figure 3. Resizing Results

Figure 3 is the result of resizing, it can be seen that the image before preprocessing has a large size of 1576 x 787, this will make it difficult for the model to perform classification because the image size used is too large. It is necessary to do resizing to reduce the size of the image by reducing its size to 150 x 150 the size used is very much different from the original size of the image. The purpose of resizing is to change the dimensions of the image to a size that suits the needs of the model. Image conversion is needed to change the image color to gray.

D. Image Conversion

Image conversion is a process of changing the image from RGB to grayscale, the image used in this study is already in grayscale form but this conversion is still carried out to prepare the image data optimally so that it allows the classification algorithm to understand and identify relevant patterns or features in the image [14]. The results of the image conversion can be seen in Figure 4.

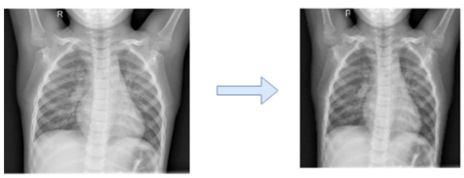

Figure 4. Image Conversion Results

Figure 4 is the result of image conversion, the initial image used is grayscale, this image conversion is still carried out for the efficiency of storage. The converted image will get a smaller size than the unconverted grayscale image. This conversion is done to improve the performance of machine learning algorithms.

E. Segmentation

Segmentation is the process of separating the image between the desired object (foreground) and the background (background) contained in an image[15]. With the segmentation process, each object in the image can be taken individually so that it can be used as input for the next process. The results of the segmentation can be seen in Figure 5.

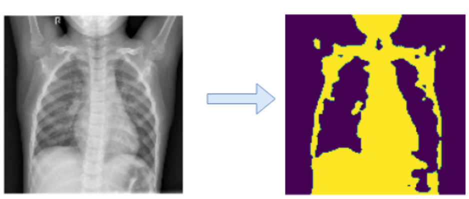

Figure 5. Segmentation Results

Figure 5 is the result of segmentation where after preprocessing consisting of resizing and image conversion, the segmentation stage is carried out. The results of the segmentation can be seen in the image above.

F. Split Data

The separation of datasets for use in the training and testing stages, known as data splitting, is a critical process in machine learning model building. In this study, the data will be split in such a way that 90% of the total data will be allocated for model training, while 10% will be allocated for model testing. Of the total 5856 data available, 90% will be used for training, which is equivalent to 5270 data. The remaining 10%, or 586 data, will be taken for testing. By ensuring that the training and testing datasets have sufficient representation of the same population, it is possible to build a model that can be expected to perform well when tested on data that has never been seen before. This step is important to avoid overfitting and validate the sustainability of the model.

G. Convolution Neural Network (CNN)

Convolutional Neural Network (CNN) is intended to process two-dimensional data with a high network depth [16]. The network in question is an artificial neural network used to process images in this case classifying and recognizing objects. CNN works by mimicking the way nerve cells communicate with interrelated neurons. CNN uses convolutional operations that apply filters in each previous input section to extract patterns and this makes CNN unique compared to other artificial neural networks [17]. CNN uses a Graphics Processing Unit (GPU) for the computational process, in other words, when using the Nvidia Cuda platform, processing can be much faster than using a Central Processing Unit (CPU) the process in the CNN algorithm can be seen in Figure 6.

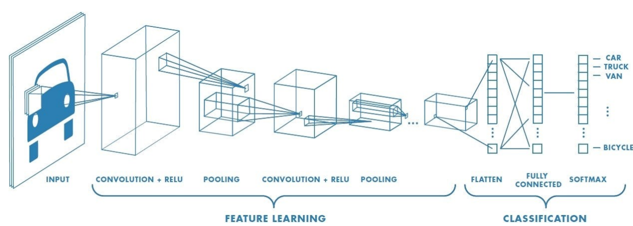

Figure 6. CNN Algorithm

Figure 6 is the CNN algorithm lag process, this process consists of two stages, namely feature learning, and classification. Input layer as matrix data from an image that is abstracted into a dual-dimensional feature map convolution layer. The output is a carnel that will be pooled. The result of pooling will be a new input layer, which will repeat the process. This is done several times to increase the number of neurons and various combinations of layer variations.

The convolutional layer is one of the core components of CNN. Its main function is to identify patterns and special features in the image data through convolution operations. The layer takes the image data and then applies a filter or kernel on top of the image, this kernel extracts the features by performing dot-product will be submitted to the next layer.

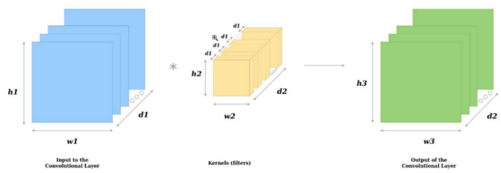

Figure 7. Convolutional Layer

Figure 7 is an image of the convolutional scanning process that occurs in the input layer using a kernel or filter from the top left corner to the bottom right corner of the input layer. The kernel is a representation of the input layer value that is run through a computational process to place the resulting value on the output layer. The input and output layers have the same dimensions, namely width, height, and depth or the number of channels [17].

The purpose of the pooling layer, a downsampling layer, is to minimize the spatial extent of the parameter count. Generally, max pooling or average pooling techniques can be used for layer pooling. The max pooling operation is performed by selecting the maximum value of the region specified in the feature map. For example, if the pooling region has a size of 2x2 then the maximum value of 4 pixels within the region will be selected as the representative value. An example of max pooling can be seen in Figure 8.

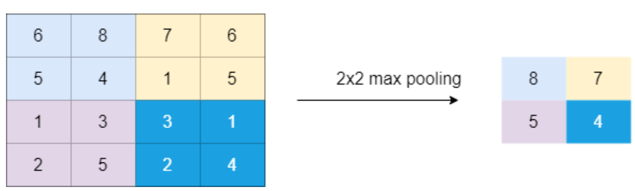

Figure 8. Max Pooling

Figure 8 is max pooling with a 4x4 layer size reduced to reduce the image to 2x2. There are several techniques used in max pooling, one of which is to find the maximum value in the layer matrix. The 4x4 layer matrix is divided into 4 layer matrices where from each layer matrix the max pool value is taken. The max pool value in the first layer is 8, the next max pool values are 7, 5, and 4.

The results of the convolution layer and final pooling in the form of a two-dimensional matrix are flattened and then inserted into the full layer. Flattening is the process of converting all values into vectors [16]. An example of the flattened process can be seen in Figure 9.

Figure 9 illustrates the result of the smoothing process, which converts the two-dimensional layers into one-dimensional vector values. At this stage, the values in the full layer will be organized as a row of one-dimensional arrays. For example, Figure 9 illustrates the transformation of matrices 8,7 and 5,4 into vectors 8,7,5,4.

Dropout is a technique in artificial neural networks used

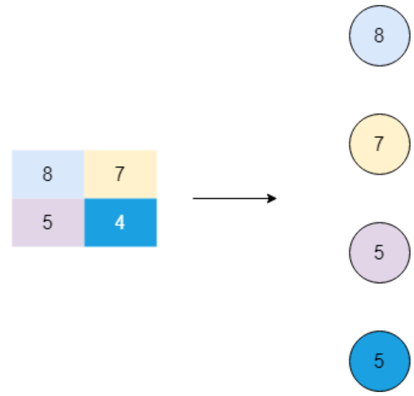

Figure 9. Flattening Process

to prevent overfitting and speed up the learning process[9]. Dropout temporarily removes or randomly discards neurons in the network. Each neuron is assigned a probability with values 0 and 1 [17]. The application of dropout can be seen in Figure 10.

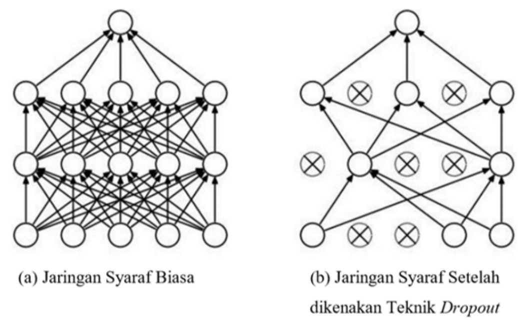

Figure 10. Application of Dropouts

Figure 10 is an example of the application of dropout. Figure 10(a) is an image of a regular neural network where the number of neurons is too many, which can cause overfitting in the training process. Dropout is done to create a new layer and will discard neurons that are considered less probable. Figure 10(b) is an artificial neural network that has been subjected to the dropout technique so that the number of neurons is less.

H. Alexnet Architecture

AlexNet architecture is one type of architecture used in convolutional neural networks (CNN) to perform image recognition. In the framework of the 2012 ImageNet Large Scale Visual Recognition Challenge, Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton created the AlexNet architecture to create a more efficient model for crushing image recognition stages [18]. GPUs are used by AlexNet to accelerate modeling and generate fast, highly accurate class predictions. Figure 11 displays the AlexNet architecture.

Figure 11 is the AlexNet architecture. The training process is carried out. After the data preparation has been processed, it will enter the input layer stage. This input

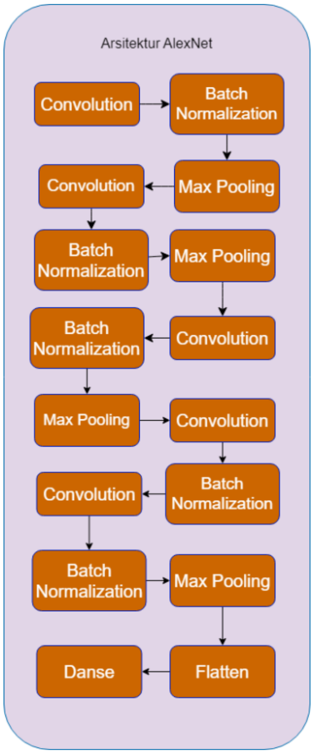

Figure 11. AlexNet Architecture

layer uses the image size (150x150x3). BatchNormalization Layer This layer is used to normalize input before going to the next Conv2D layer. MaxPool2D Layer This layer is used to extract important features, reduce data dimensions, and increase invariance to shifts, all of which are important factors in building efficient and accurate models. The size used is 3x3 pooling with 2x2 steps. Conv2D Layer This layer is used to form a strong basis for the extraction and recognition of important features in images. This layer uses 256 filters with a kernel size of 5x5 and a stride of 1x1. The activation function used is ReLU, and this layer is equipped with padding. BatchNormalization Layer This layer is used to normalize input before going to the next Conv2D layer. Flatten Layer This layer is used to produce multi-dimensional input into one-dimensional input for a fully connected layer. Dense Layer This layer is used to assist in the extraction of abstract features, classification, regression, and understanding patterns in data, this layer uses 2 neurons and a softmax activation function to generate class predictions. This model uses Keras (TensorFlow) to perform image recognition using several layers of Conv2D, MaxPooling2D, BatchNormalization, and Dense. This model also uses ReLU and Dropout activation functions to improve image recognition performance and accuracy.

The AlexNet model testing process can be seen in Figure 11 for the AlexNet architecture. After getting the output of the training iteration. In the testing stage, a model evaluation will be carried out, and in the final stage of the testing process, classification results will be obtained which will receive an accuracy value.

I. GoogleNet Architecture

Google researchers created the deep convolutional neural network architecture known as GoogleNet. The Inception module, which is intended to expand the network's breadth and depth without appreciably raising its parameter count, is what distinguishes the GoogleNet architecture [19]. The Inception module consists of several convolutional filters of different sizes, which are applied in parallel to the same input. The outputs of these filters are then combined and passed on to the next layer. The GoogleNet architecture can be seen in Figure 12.

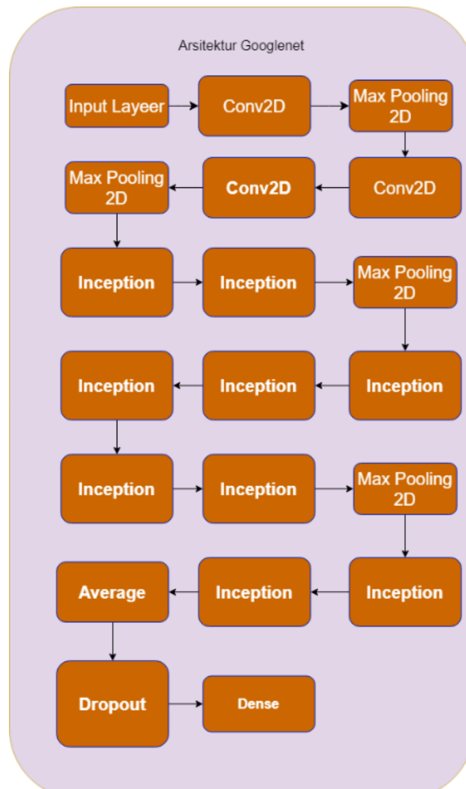

Figure 12. GoogleNet Architecture

Figure 12 is the GoogleNet architecture, the training stage is the process of data that has been prepared will be entered into the GoogleNet architecture, where the input layer is the initial stage in the GoogleNet architecture, this layer receives input (150x150x3) pixels. Three Conv2D layers with filter 64 kernel-size 7 x 7, and strides 2. Conv2D is used for. After the convolution process, the resulting feature map is more compact. Two layers of MaxPooling2D with pool size 3 x 3 and strides 2. This aims to produce a more compact feature map and remove unimportant details. Each layer of the Inception block combines multiple Conv2D layers with different filters and the same kernel size. Inception block aims to reduce computational load and increase data depth. Average is used to combine the results of the previous layers into a scale of 1 x 1 x 2. Dropout is used to reduce overfitting by randomly removing some neurons. Dense This layer generates class predictions using a softmax activation function.

Figure 12 shows the GoogleNet testing procedure. This procedure is run after the training phase, where model testing is performed on 10% of the data. The test results are presented as a confusion matrix, which determines the accuracy value of the testing procedure.

J. *Model Evaluation*

The testing of the model design that has been made is by obtaining the accuracy value. The results of the testing model displayed by the confusion matrix are tested to determine the performance of CNN in the classification of pneumonia diseases. The confusion matrix helps to understand the extent to which the model is successful or unsuccessful in performing classification [20].

Figure 13. Confusion Matrix

Figure 13 shows the data obtained based on the confusion matrix system testing there are True Positive (TP), True Negative (TN), False positive (FP), and False Negative (FN) values. From the confusion matrix, we can calculate precession, recall, F1-score, and accuracy.

Precision is used to calculate how many cases are predicted as positive by the model that should be predicted as positive, following the precision equation.

4. Result and Discussion

The results of this study use Python tools using pneumonia image data. The image data is divided into 90% training data and 10% testing data. The results of the study discuss the analysis of the comparison of pneumonia image classification with GoogleNet and AlexNet architectures.

A. *Training Model*

This training uses Adam optimization which helps the model in training. Adam optimization is one of the methods used to calculate the learning rate for each different parameter. The dataset used in this study is 5856 roentgen thorax data with such a large amount of data, it takes a long time to train the data simultaneously because the memory in the computer has limitations, this requires parameters such as batch size and epoch. Batch size is a parameter that divides the dataset into several groups. The model training process is carried out with Epoch 20 where each model will iterate 20 times There are other parameters carried out in this study such as the ReduceLROnPlateau parameter which will reduce the learning rate if the learning rate at the epoch causes the model to not show progress towards what is learned. After the mode training is completed, the results of each model experiment are stored in .h5 format so that they can be retrieved later in training. The results of the training dataset can be seen in Table III.

TABLE III. Classification Results Of Googlenet Model

Epoch	GoogleNet			AlexNet		
	Time (s)	Accuracy	Training Loss	Time (s)	Accuracy	Training Loss
1	245	0.2736	0.8366	255	0.8151	2.0417
2	186	0.7297	0.5834	210	0.8912	0.4067
3	189	0.7529	0.4878	206	0.9353	0.1684
4	188	0.8360	0.3570	210	0.9427	0.1525
5	183	0.8828	0.2845	213	0.9481	0.1442
6	848	0.8906	0.2731	206	0.9498	0.1350
7	176	0.9199	0.2081	208	0.9597	0.1108
8	174	0.9355	0.1728	211	0.9694	0.0790
9	175	0.9403	0.1490	207	0.9766	0.0660
10	900	0.9608	0.0996	208	0.9793	0.0554
11	178	0.9703	0.0774	21	0.9764	0.0658
12	172	0.9684	0.0737	209	0.9882	0.0337
13	1348	0.9755	0.0627	211	0.9905	0.0269
14	171	0.9806	0.0535	208	0.9935	0.0201
15	172	0.9899	0.0277	208	0.9937	0.0189
16	171	0.9935	0.0182	207	0.9962	0.0105
17	425	0.9966	0.0116	208	0.9977	0.0086
18	187	0.9977	0.0102	210	0.9981	0.0079
19	190	0.9975	0.0084	225	0.9975	0.0073
20	140	0.9977	0.0083	218	0.9985	0.0062

Table III Training Result is the result of training data with 20 iterations, there are columns of accuracy and loss time. the first model used by GoogleNet iteration length is 245/s with an accuracy value of 0.2736 with a loss of 0.8366 this result is relatively low because the GoogleNet model learns new data so to increase the iteration value must learn the next data given. In the second iteration, the accuracy rose to 0.7297 and loss to 0.5834, this can be interpreted that Google Net experienced a significant increase in accuracy of 0.4561 and can be interpreted that Google Net can increase accuracy and reduce the loss required in iterations.

For the AlexNet model, the initial old iteration 255/s gets an accuracy value of 0.8151 this result is quite low compared to the next iteration in AlexNet. The second iteration gets an accuracy of 0.8912 with a loss value of 0.4067 this shows that AlexNet can learn to increase accuracy and reduce loss. After running 20 iterations in the GoogleNet column there are 2 highest accuracy values of 0.9977 with a time of 187 / s loss value of 0.0102 and with a time of 140 / s loss of 0.0083. this shows that GoogleNet can learn the data given well. the AlexNet column has the highest accuracy value of 0.9985 with a time of 218 / s loss of 0.0062. this can be interpreted that AlexNet has a higher accuracy even though the accuracy value is only 0.0008 different.

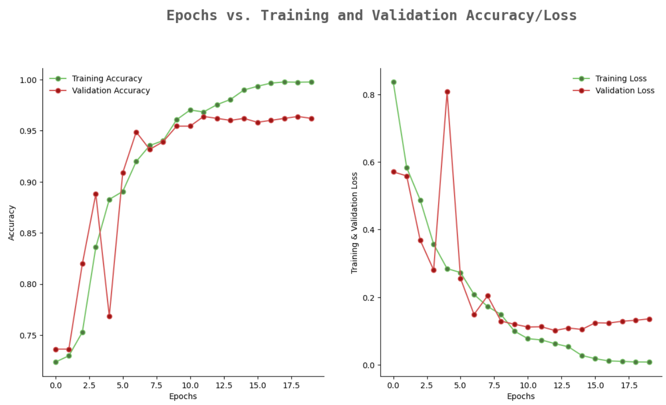

Figure 14. Accuracy and loss graph of GoogleNet training

Figure 14 is a graph of GoogleNet training accuracy and loss, it can be seen from the graph that the accuracy of GoogleNet in the first iteration is very low and has increased in subsequent accuracy. The loss value for each iteration decreased, proving that the model can reduce errors in learning the pneumonia dataset. This proves that GoogleNet can learn the dataset well.

The red line from Figure 14 is the validation used in this study. The validation technique used is a split validation set, where the training dataset is used 10% for validation data. This validation is to tune the meters used and can avoid overfitting. This validation performance is processed after the training data has been processed.

Figure 15 is a graph of AlexNet training accuracy and loss, it can be seen based on the graph that the accuracy of AlexNet in the first iteration is very low and has increased in subsequent accuracy. The loss value for each iteration decreased, proving that the model can reduce errors in learning the pneumonia dataset. This proves that AlexNet can learn the dataset well.

B. Testing Model

The dataset used is 90% for training the model and 10% for testing the model, testing this model aims to determine the accuracy level of the model used. Images were used

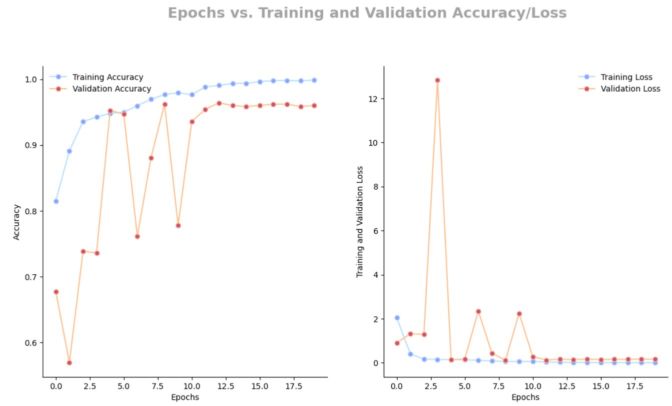

Figure 15. Accuracy and loss graph of GoogleNet training

for testing 162 normal images and 424 pneumonia images. The results of the classification are visualized in the form of confusion matrix models AlexNet and GoogleNet.

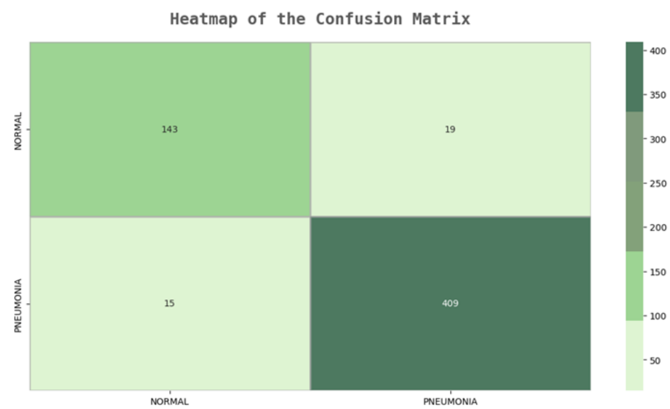

Figure 16. GoogleNet confusion matrix results

Figure 16 is the result of the confusion matrix GoogleNet, it can be seen that the results of the confusion matrix of GoogleNet test results with a total of 5856 data divided into 90% training data and 10% testing data. 10% of 5856 is 586 images used for testing the model. The test results on the confusion matrix are visualized in Table IV.

TABLE IV. Classification Results Of Googlenet Model

Index	Class	Classification True	Total Data	Accuracy (%)
0	Normal	148	162	89
1	Pneumonia	409	424	96

Table IV shows that the highest classification result is in the pneumonia class with an accuracy of 96% of images successfully classified by class. While normal accuracy has an accuracy of 89%. The data obtained from the confusion matrix is entered into the accuracy formula to get the accuracy value.

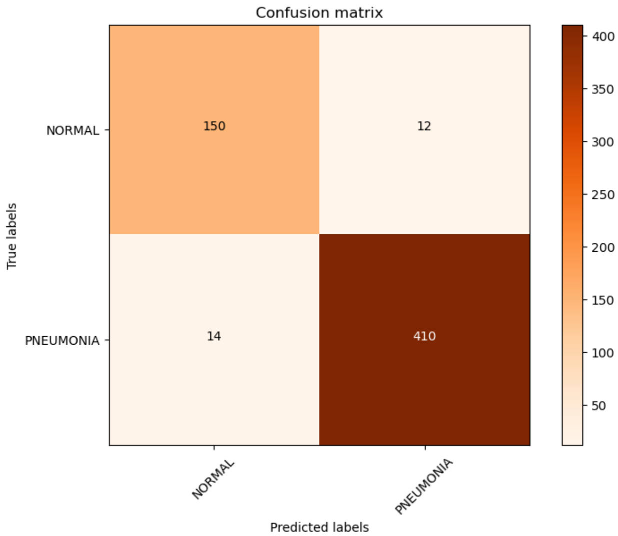

Figure 17. Alexnet confusion matrix results

Figure 17 displays the results of the confusion matrix of the test results from AlexNet with a total of 5856 data divided into 90% training data and 10% testing data. 10% of 5856 is 586 images used for testing the model. The test results on the confusion matrix on the AlexNet model are visualized in Table V.

TABLE V. AlexNet Model Classification Results

Index	Class	Classification True	Total Data	Accuracy (%)
0	Normal	150	162	92
1	Pneumonia	410	424	97

Table V shows that the highest classification result is in the pneumonia class with an accuracy of 97% of images successfully classified by class. While normal accuracy has an accuracy of 92%. The data obtained from the confusion matrix is entered into the accuracy formula to get the accuracy value.

5. Conclusions and Future Work

In this study, pneumonia classification was conducted using GoogleNet and AlexNet architectures. From the research that has been done, it can be concluded that the AlexNet architecture has a higher accuracy than the GoogleNet accuracy. The accuracy of AlexNet is 96% and the accuracy of GoogleNet is 94%. However, both of these architectures are highly recommended for the classification of rotgen thorax pneumonia images. However, this research still has shortcomings where the data used is still small and the preprocessing techniques used are not perfect so the results cannot achieve higher accuracy or may reach 100%. From the results obtained, it can be concluded that Alexnet is better than GoogleNe in the classification of Pneumonia medical images.

References

[1] D. Varshni, K. Thakral, L. Agarwal, R. Nijhawan, and A. Mittal, "Pneumonia detection using cnn-based feature extraction," in *Proc. 2019 3rd IEEE Int. Conf. Electr. Comput. Commun. Technol. ICECCT 2019*, 2019, pp. 1–7.

[2] N. Chebib *et al.*, "Pneumonia prevention in the elderly patients: the other sides," *Aging Clin. Exp. Res.*, vol. 33, no. 4, pp. 1091–1100, 2021.

[3] J. Yopento, E. Ernawati, and F. F. Coastera, "Identifikasi pneumonia pada citra x-ray paru-paru menggunakan metode convolutional neural network (cnn) berdasarkan ekstraksi fitur sobel," *Rekursif J. Inform.*, vol. 10, no. 1, pp. 40–47, 2022.

[4] H. Ren *et al.*, "Interpretable pneumonia detection by combining deep learning and explainable models with multisource data," *IEEE Access*, vol. 9, pp. 95 872–95 883, 2021.

[5] D. M. Wonohadidjojo, "Perbandingan convolutional neural network pada transfer learning method untuk mengklasifikasikan sel darah putih," *Ultim. J. Tek. Inform.*, vol. 13, no. 1, pp. 51–57, 2021.

[6] A. U. Ibrahim, M. Ozsoz, S. Serte, F. Al-Turjman, and P. S. Yakoi, "Pneumonia classification using deep learning from chest x-ray images during covid-19," *Cognit. Comput.*, no. 0123456789, 2021.

[7] H. R. Burhani, I. Fitri, and A. Andrianingsih, "Perbandingan naïve bayes dan certainty factor pada sistem pakar untuk mendiagnosa dini penyakit glaukoma," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 5, no. 3, p. 291, 2020.

[8] N. Abdillah, A. K. W. Hapantenda, A. Habib, and I. Listiowarni, "Klasifikasi viral pneumonia mengunakan metode convolutional neural network dan support vector machine," *Konvergensi*, vol. 18, no. 2, pp. 50–56, 2022.

[9] N. Nurkhasanah and M. Murinto, "Klasifikasi penyakit kulit wajah menggunakan metode convolutional neural network," *Sainteks*, vol. 18, no. 2, pp. 183–190, 2022.

[10] J. Gatc and F. Maspiyanti, "Prediksi parasit plasmodium pada citra mikroskopis sel darah merahdengan convolutional neural networks," *J. Buana Inform.*, vol. 13, no. 1, pp. 31–41, 2022.

[11] Z. Gong and L. Kan, "Segmentation and classification of renal tumors based on convolutional neural network," *J. Radiat. Res. Appl. Sci.*, vol. 14, no. 1, pp. 412–422, 2021.

[12] L. B. Tribuzy, Y. P. Torres, R. S. Furtado, L. C. G. Júnior, N. P. Bitar, and W. S. Júnior, "Vehicle license plate preprocessing techniques using graphical interface," in *2020 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-Taiwan)*. IEEE, 2020, pp. 1–2.

[13] A. J. Rindengan and M. Mananohas, "Perancangan sistem penentuan tingkat kesegaran ikan cakalang menggunakan metode curve fitting berbasis citra digital mata ikan," *J. Ilm. Sains*, vol. 17, no. 2, p. 161, 2017.

[14] C. Saravanan, "Color image to grayscale image conversion," in *2010 2nd Int. Conf. Comput. Eng. Appl. ICCEA 2010*, vol. 2, 2010, pp. 196–199.

[15] R. T. Wahyuningrum *et al.*, "Segmentasi citra x-ray dada menggunakan metode modifikasi deeplabv3+," *J. Tek. Inf. dan Komunikasi*, vol. 10, no. 3, pp. 687–698, 2023.

[16] 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.

[17] S. Sunardi, A. Fadlil, and D. Prayogi, "Sistem pengenalan wajah pada keamanan ruangan berbasis convolutional neural network," *Jurmal Sains Komput. Inform.*, vol. 6, no. 2, pp. 636–647, 2022.

[18] H. C. Chen *et al.*, "Alexnet convolutional neural network for disease detection and classification of tomato leaf," *Electron.*, vol. 11, no. 6, pp. 1–17, 2022.

[19] M. F. Aslan, K. Sabanci, A. Durdu, and M. F. Unlersen, "Covid-19 diagnosis using state-of-the-art cnn architecture features and bayesian optimization," *Comput. Biol. Med.*, vol. 142, no. October 2021, p. 105244, 2022.

[20] P. Garg, N. Sharma, Sonal, and B. Shukla, "Predicting the risk of cardiovascular diseases using machine learning techniques," *Int. J. Intell. Syst. Appl. Eng.*, vol. 11, no. 2s, pp. 165–173, 2023.

Rio Subandi received his bachelor's degree from the Department of Informatics Engineering, Faculty of Industrial Technology, Ahmad Dahlan University, Yogyakarta, Indonesia in 2021, and his master's degree in informatics in 2024. His research interests include artificial intelligence, deep learning and machine learning.

Herman is a lecturer at the Master of Informatics at Ahmad Dahlan University, Yogyakarta. He studied Bachelor's degree at the Perbanas Institute of Information Engineering, Jakarta, then continued his Master's and Doctoral studies at Computer Science Universiti Teknologi Malaysia. His area of expertise is Mobile and Multimedia Technologies.

Anton Yudhana holds the academic rank of Professor in Electrical Engineering. He obtained his Bachelor of Electrical Engineering from Institut Teknologi Sepuluh Nopember in 2001, his Master of Electrical Engineering from Universitas Gadjah Mada in 2005, and his Doctorate from Universiti Teknologi Malaysia in 2011. Currently, he has been a permanent lecturer at the University of Ahmad Dahlan (UAD) since then, and he serves as the leader at the Centre for Electrical and Electronic Research and Development (CEERD). His research interests include precision agriculture, biomedical engineering, and IoT-based food engineering.