Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 23, No. 1, November 2023, pp. 17∼28
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v23i1.3197

Recognizing Pneumonia Infection in Chest X-Ray Using Deep Learning

Ni Wayan Sumartini Saraswati[1], I Wayan Dharma Suryawan[1], Ni Komang Tri Juniartini[1], I Dewa Made Krishna Muku[1], Poria Pirozmand[2], Weizhi Song[3]
[1]Institut Bisnis dan Teknologi Indonesia, Denpasar, Indonesia
[2]Holmes Institute Sydney, Australia
[3]University of Science and Technology, Hong Kong, China

Article Info

Article history:

Received August 24, 2023
Revised September 18, 2023
Accepted October 10, 2023

Keywords:

Chest X-Ray
Convolution Neural Network
Deep Learning
Image Classification
Pneumonia

ABSTRACT

One of the diseases that attacks the lungs is pneumonia. Pneumonia is inflammation and fluid in the lungs making it difficult to breathe. This disease is diagnosed using X-Ray. Against the darker background of the lungs, infected tissue shows denser areas, which causes them to appear as white spots called infiltrates. In the image processing approach, pneumonia-infected X-rays can be detected using machine learning as well as deep learning. The convolutional neural network model is able to recognize images well and focus on points that are invisible to the human eye. Previous research using a convolutional neural network model with 10 convolution layers and 6 convolution layers has not achieved optimal accuracy. The aim of this research is to develop a convolutional neural network with a simpler architecture, namely two convolution layers and three convolution layers to solve the same problem, as well as examining the combination of various hyperparameter sizes and regularization techniques. We need to know which convolutional neural network architecture is better. As a result, the convolutional neural network classification model can recognize chest x-rays infected with pneumonia very well. The best classification model obtained an average accuracy of 89.743% with a three-layer convolution architecture, batch size 32, L2 regularization 0.0001, and dropout 0.2. The precision reached 94.091%, recall 86.456%, f1-score 89.601%, specificity 85.491, and error rate 10.257%. Based on the results obtained, convolutional neural network models have the potential to diagnose pneumonia and other diseases..

Corresponding Author:

Ni Wayan Sumartini Saraswati, +628124653960
Faculty of Technology and Informatics, Department of Informatics Engineering,
Institut Bisnis dan Teknologi Indonesia, Denpasar, Indonesia,
Email: sumartini.saraswati@gmail.com

How to Cite:
N. W. Saraswati, I. W. Suryawan, N. K. Juniartini, and I. D. M. Muku, ”Recognizing Pneumonia Infection in Chest X-Ray Using Deep Learning”, *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 23, no. 1, pp. 17-28, Oct. 2023.

1. INTRODUCTION

In the medical world, X-rays are used in examinations that can show the inside of the human body. Technically, the X-ray process requires no preparation or aftercare for the patient. The procedure is also quick and painless. When an X-ray machine creates an image, the energy from the X-rays (the name refers to the type of radiation used) travels through the body and is absorbed by various bodies at different speeds, such as bones, organs, and muscles. Then, detectors on the other side of the body collect the X-rays as they pass through them and turn them into images. Solid-body parts will appear white like bone because radiation cannot pass through solid tissue and bone. On the other hand, in softer tissues such as the lungs, X-rays pass through with little problem and will show up as darker areas. X-rays are effective in visualizing the inside of the body, and from these images, doctors can detect or diagnose related conditions, such as fractures, heart conditions, and even pneumonia. In pneumonia, the infected tissue shows a denser area, so it will see white spots on the darker background of the lungs, and these white spots are known as infiltrates. Pneumonia itself has become the main cause of infant death worldwide. Even in Indonesia, pneumonia is the second-most common cause of infant and toddler mortality [1]. In 2018, UNICEF revealed 800,000 under-five deaths yearly due to pneumonia [2]. The World Health Organization (WHO) also estimates that pneumonia accounts for 15% of deaths in children under five years [3].

In this approach, image processing can be used in the healthcare industry, particularly in the area of disease identification [4]. Chest X-rays are image data, so they can be used as input to build a pneumonia detection model; an appropriate method is needed because it is image data. One very popular method for detecting patterns in image data is the machine learning method [4–6]. In its development, the performance of machine learning methods is compared with deep learning methods. In certain cases, the image recognition method with deep learning can even recognize images that normal eyes cannot capture. Research conducted by [7] detects leaf disease images and compares the performance of machine learning with deep learning. The results reveal that the method in deep learning achieves higher accuracy than using machine learning. Likewise, image recognition research by [8] [9] shows that deep learning methods provide better performance than machine learning methods. Deep learning is an evolution of machine learning technique. This method has been proven capable of handling various cases in image processing and object detection [10]. The deep learning method in image processing provides the advantage that it does not require feature extraction methods before classification and is effective for large amounts of data. A popular deep learning method used for image detection is the Convolutional Neural Network (CNN), one of the neural network techniques. CNN is frequently utilized to process picture data [11]. The research by [12] examined the images of brain tumors, classified into two categories, glioma, and meningioma, using CNN with the EfficientNet-B3 architecture. The results of this study obtained high accuracy, reaching 99.7%.

According to [13], if properly trained, CNN can detect images by focusing on points that are not visible to the human eye. Information in the image can be processed well by CNN because the system is like image recognition in the human visual cortex [14]. Research conducted by [15] compared six deep-learning models. A simple CNN was used with data augmentation, and a simple CNN without data augmentation was used to detect pneumonia. This model uses 10-layer convolution, but the accuracy results obtained are still not optimal, namely less than 80%. Another study that detected pneumonia using CNN was conducted by [3]. This research used six convolution layers, which were not optimal because the training accuracy obtained was less than 80%.

The difference between this research and previous research that this research proposes pneumonia detection using CNN on X-ray images with simpler architectural layers but is expected to obtain more optimal accuracy results. With a simpler layer, the computing process is expected to be lighter. The novelty of this research is that we make model of two architectures, namely CNN with two convolution layers and CNN with three convolution layers. These two architectures were then implemented using varying hyperparameter sizes and regularization to find the best combination of measures for detecting pneumonia. The purpose of this study is to find the optimal CNN architecture to solve the problem of detecting chest X-rays infected with pneumonia. Detection is carried out in the form of a classification of two classes, namely the class of images infected with pneumonia and the class of images in normal conditions. The contribution of this research is reducing computational costs of deep learning method in recognizing pneumonia infections for lung X-ray images. This is supported by the 3 layer CNN architecture which is simpler than previous research but with the right hyperparameters it can provide better performance.

2. RESEARCH METHOD

This study detects pneumonia using the deep learning method with the CNN algorithm. This research is an engineering research method to obtain criteria following predetermined requirements by applying science to a design. This research is divided into several stages, starting from data collection, data-preprocessing, model building, testing, and model evaluation, and describes several experimental scenarios. In this study, the steps that will be carried out to classify images of lungs infected with pneumonia are described through an experimental flowchart shown in Figure 1 and described in the following sub-chapters.

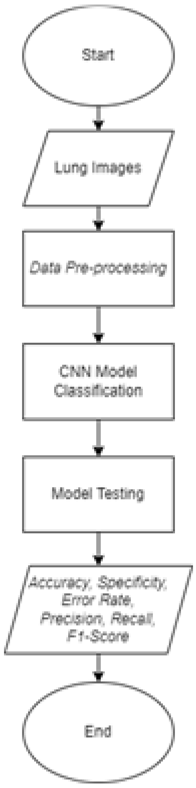


Figure 1. CNN model experiment flowchart

2.1. Data Source

This research was conducted starting with collecting datasets as input for the classification of lungs infected with pneumonia. Learning by deep learning is guided learning in which training data is prepared to build a model, and then the model is tested on test data that has been prepared. Because the COVID-19 X-ray depicts pneumonia with the presence of ground glass opacity (GGO) [16], the dataset of this study is represented using COVID-19 X-ray images and normal X-rays. Images of lungs infected with COVID-19 were downloaded from research by [17], and images of the lungs under normal conditions from the Ganesha General Hospital, Gianyar, Bali [6]. The collected image data has been validated and verified by experts. A total of 165 images with lungs infected with COVID-19 totaling 82 images and lungs with normal conditions totaling 83 images. Lung image in jpg extension. Figure 2 illustrates a chest X-ray with COVID-19 infection, while Figure 3 shows a typical chest X-ray image. This work uses chest X-ray pictures of normal and COVID-19-infected lungs equally as training and test data.

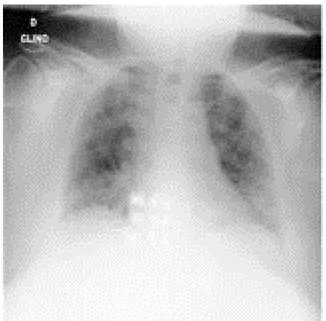


Figure 2. Lungs infected with COVID-19

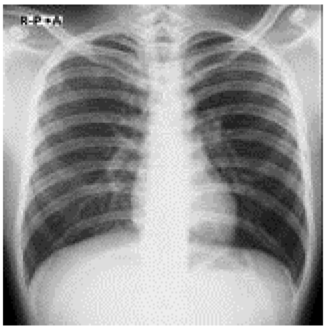


Figure 3. Lungs with normal condition

2.2. Data Preprocessing

Data preprocessing is the first step that is carried out before the model training process. Data preprocessing will convert the raw image data into the appropriate format required as model input. The data preprocessing stage aims to remove irrelevant data so that the modeling process becomes more optimal. The image will go through cropping, resizing, and converting into a grayscale image. In classification, computers recognize images differently from how humans recognize images. The image is recognized in the form of a pixel array. For example, in an image that is 160x160 in size, the array size is 160x160x3, where the first 160 is the width, the second 160 is the height, and 3 is the RGB value. If the value is 160x160x1, then the number 1 is the image for grayscale. Values in the image have a range of 0-255, which describes the intensity of the pixels at each point [18].

After the image has gone through the data preprocessing process, the next step is to build a classification model using the CNN method. This classification process goes through two stages, namely, the process of training (training) and testing (testing). The dataset is split in this study by a 90%:10% ratio. The dataset will be split between training and testing portions, with training data making up 90% of the dataset.

Classification models using data augmentation techniques help increase data diversity during the model training process, especially for limited data. This research will use several types of data augmentation techniques, such as rotation to rotate the image, shift to lift the image both horizontally and vertically, flip to flip the image, shear to shift the image, zoom to enlarge the image, brightness to increase the brightness of the image, and rescale used before image augmentation which serves to normalize the image.

2.3. CNN Model

Convolutional Neural Network (CNN) is a technique created from the Multi-Layer Perceptron (MLP) algorithm, so it works similarly to MLP. In MLP, each neuron is one-dimensional, unlike CNN, where each neuron is two-dimensional. Therefore, CNN is widely applied to two-dimensional images, as shown in Figure 4 [19].

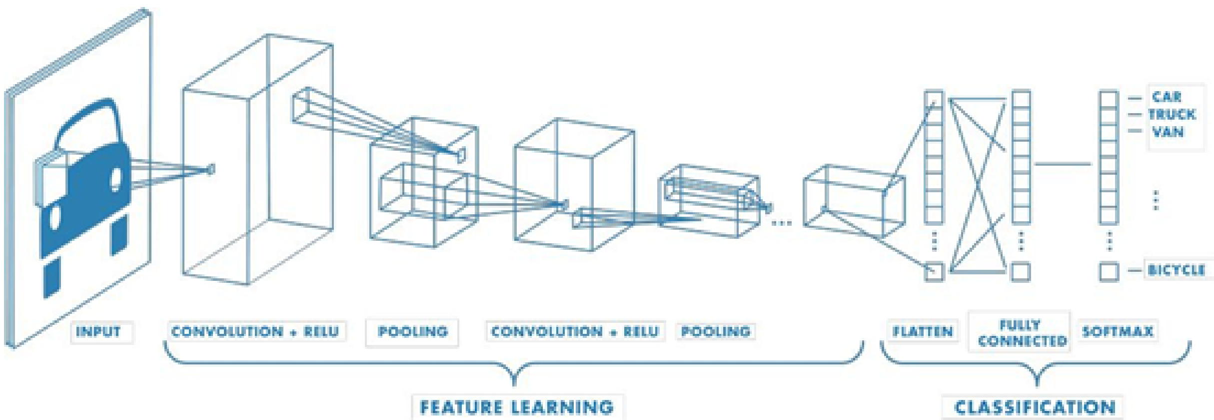


Figure 4. CNN architecture [20]

The CNN model will go through two stages: feature learning and classification. The feature learning process is translating images into features in the form of numbers. This figure explains the image in the form of a feature map. The feature map is presented as a multi-dimensional array. At the feature learning stage, CNN will go through the convolution layer, activation function, and pooling layer process. Convolution is a multiplication operation between two matrices whose results will be added together. This convolution produces a feature map. After being convolved, max pooling is carried out on the pooling layer, which aims to reduce matrix size, control overfitting, and calculate in the network. The activation functions used are ReLu and Sigmoid. Dropouts are also used to prevent overfitting.

In the classification process, CNN will use the fully connected layer. A feature vector created from the feature map is then input from the fully connected layer [21]. This conversion is called the flattening technique. The result of this flattening will be one neuron [22]. The CNN model can be described in this study in the Figure 5 flow below.

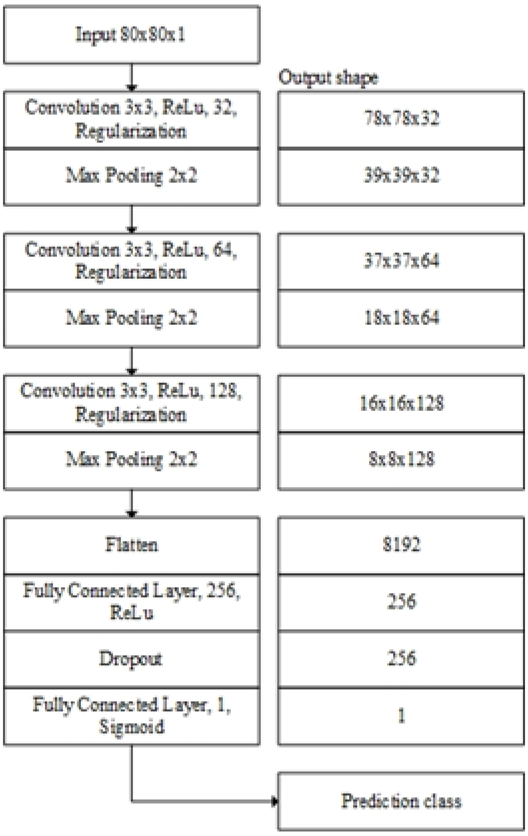


Figure 5. CNN model flowchart

The input image will undergo a feature learning process with three convolution layers, two max-pooling layers, and one dropout layer. The input image used is 80x80x1 in size. Number one is the channel image, which is a grayscale image. This study uses two models of architecture. The first architecture model was the main model, and the second architecture was used as a comparison.

The first model architecture consists of three convolution layers; namely, the first convolution uses a filter value of 32, a 3x3 kernel, regularization, and ReLu activation. The value used by max pooling is 2x2. The second convolution employs a 3x3 kernel, regularization, activation of ReLu, and max pooling with a filter value of 64. The third convolution employs a 3x3 kernel, ReLu activation, and max pooling with a filter value of 128. Entering the classification process, the convolution results will go through a flattening process, a fully connected layer with 256 neurons, dropout, and sigmoid activation. The second model uses two layers, which only consist of filters 32 and 64, while the fully connected layer uses 128 neurons. Figure 6 shows the main architecture.

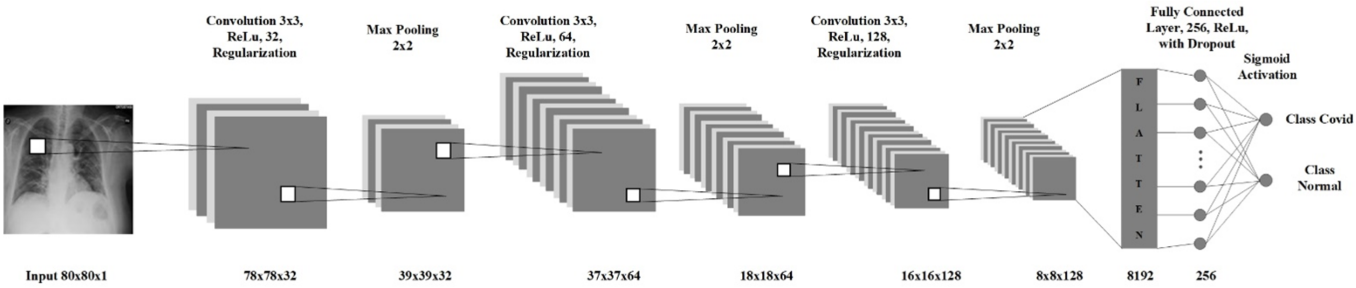


Figure 6. Main CNN model architecture

2.4. Model Testing and Evaluation

In order to maximize accuracy outcomes, the built-in model will be validated using the K-Fold Cross Validation technique. This method randomly divides the dataset into many parts, processes each partition K times or k-folds, and uses the first K partitions as test data and the remaining partitions as training data for each experiment. Cross-validation will be used tenfold in this project. The partition will be split into 10 partitions, 10 for testing data and 10 for training data. Figure 7 shows how to divide the data into 10 folds.

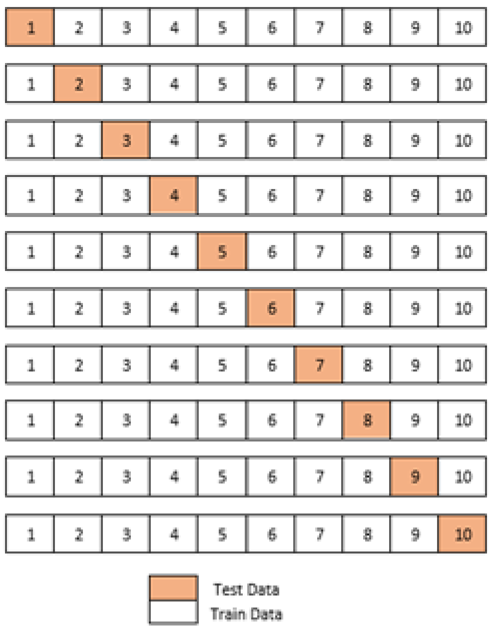


Figure 7. 10-fold cross-validation

Next, the model performance results were measured using a confusion matrix. The confusion matrix functions as a table and can distinguish between various tuples. Evaluation of this model is usually used to measure the classification of two classes or binary. Recognized results use True Positives and False Positives to provide correct predicted classification information, and then there are True Negatives and False Negatives to provide incorrectly predicted information. Table 1 shows the confusion matrix.

Table 1. Confusion Matrix

	Predicted Class	
Actual Class	True Positive (TP)	False Negative (FN)
	False Positive (FP)	True Negative (TN)

Information:

1. True Positive (TP) data is data that is positive and is expected to be positive.
2. False Positive (FP) refers to negative data that is expected to be positive.
3. True Negative (TN) data is information that is anticipated to be negative.
4. False Negative (FN) data is positive information that is expected to be negative.

This confusion matrix will calculate accuracy, precision, recall, specificity, error rate, and f1-score. The accuracy demonstrates how precisely the model can be correctly categorized. Recall displays the proportion of accurately positive predictions to all correctly positive data. The precision metric displays the proportion of correctly predicted positive outcomes to positive predictions. A weighted average of recall and precision is the F1-Score. The ratio of unfavorable forecasts to overall unfavorable data is called specificity. The error rate shows the predicted error rate in the model.

2.5. CNN Scenarios

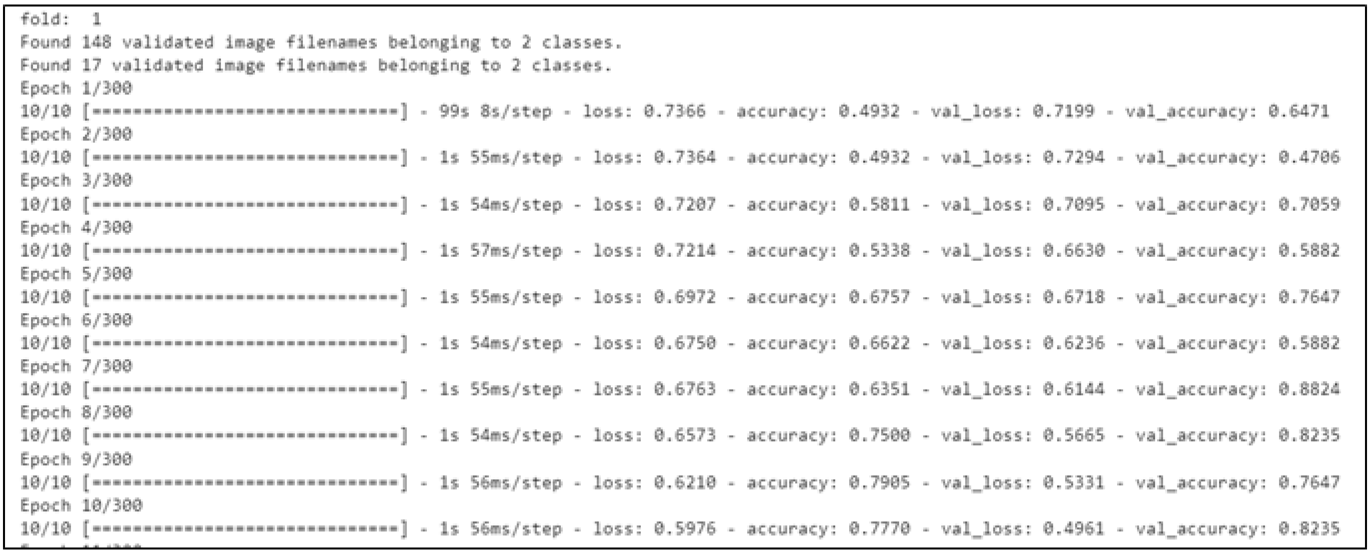
In this study, both architectures use Adam optimization. Adam optimization is an optimization method used to find optimal weights, maximize accuracy, and minimize errors or losses so that the model can predict classes accurately. Then, both architectures will use several hyperparameter sizes, such as epochs with values 50, 100, 150, 200, 250, and 300. Epoch is a number that determines how many times the algorithm works through the entire training dataset. Batch size is the number of samples that must be worked on before updating parameters in the internal model, where sizes 16 and 32 are used in the model. Then, a learning rate with a size of 0.0001 is used, where the learning rate is used to calculate the connection weight value. The regularization techniques used are L2 regularization and dropout. The regularization method is to control the complexity of the model. This method is used to reduce model error. In the model, L2 is used with a size of 0.001 and L2 0.0001. Dropout prevents overfitting and speeds up the learning process, where the model uses sizes 0.2 and 0.3. Based on this, the total number of models built is 96 models. Table 2 shows the scenarios with each type of architecture and their hyperparameter sizes, so that based on the two types of architecture and their hyperparameter sizes and regularization techniques, it produces a combination of 16 scenarios that will be tried on the CNN model.

Table 2. Experiment Scenario

Scenario	Batch Size	Architecture	Regularization L2	Dropout
1	16	2 convolution layers	0.0001	0.2
2	16	2 convolution layers	0.0001	0.3
3	16	2 convolution layers	0.001	0.2
4	16	2 convolution layers	0.001	0.3
5	32	2 convolution layers	0.0001	0.2
6	32	2 convolution layers	0.0001	0.3
7	32	2 convolution layers	0.001	0.2
8	32	2 convolution layers	0.001	0.3
9	16	3 convolution layers	0.0001	0.2
10	16	3 convolution layers	0.0001	0.3
11	16	3 convolution layers	0.001	0.2
12	16	3 convolution layers	0.001	0.3
13	32	3 convolution layers	0.0001	0.2
14	32	3 convolution layers	0.0001	0.3
15	32	3 convolution layers	0.001	0.2
16	32	3 convolution layers	0.001	0.3

3. RESULT AND ANALYSIS

This research uses Python as a programming language to build models and uses several supporting libraries, such as TensorFlow and Keras. In addition, for preprocessing such as cropping, resizing, and others, OpenCV is used to process it. In this study, the model was built through a training process after going through the data preprocessing process. The training process on the CNN method displays epochs, time of each epoch, training accuracy, loss of accuracy, accuracy of validation, and loss of validation. Figure 8 shows the training process carried out on the CNN model on one of the folds.



```
fold:  1
Found 148 validated image filenames belonging to 2 classes.
Found 17 validated image filenames belonging to 2 classes.
Epoch 1/300
10/10 [==============================] - 99s 8s/step - loss: 0.7366 - accuracy: 0.4932 - val_loss: 0.7199 - val_accuracy: 0.6471
Epoch 2/300
10/10 [==============================] - 1s 55ms/step - loss: 0.7364 - accuracy: 0.4932 - val_loss: 0.7294 - val_accuracy: 0.4706
Epoch 3/300
10/10 [==============================] - 1s 54ms/step - loss: 0.7207 - accuracy: 0.5811 - val_loss: 0.7095 - val_accuracy: 0.7059
Epoch 4/300
10/10 [==============================] - 1s 57ms/step - loss: 0.7214 - accuracy: 0.5338 - val_loss: 0.6630 - val_accuracy: 0.5882
Epoch 5/300
10/10 [==============================] - 1s 55ms/step - loss: 0.6972 - accuracy: 0.6757 - val_loss: 0.6718 - val_accuracy: 0.7647
Epoch 6/300
10/10 [==============================] - 1s 54ms/step - loss: 0.6750 - accuracy: 0.6622 - val_loss: 0.6236 - val_accuracy: 0.5882
Epoch 7/300
10/10 [==============================] - 1s 55ms/step - loss: 0.6763 - accuracy: 0.6351 - val_loss: 0.6144 - val_accuracy: 0.8824
Epoch 8/300
10/10 [==============================] - 1s 54ms/step - loss: 0.6573 - accuracy: 0.7500 - val_loss: 0.5665 - val_accuracy: 0.8235
Epoch 9/300
10/10 [==============================] - 1s 56ms/step - loss: 0.6210 - accuracy: 0.7905 - val_loss: 0.5331 - val_accuracy: 0.7647
Epoch 10/300
10/10 [==============================] - 1s 56ms/step - loss: 0.5976 - accuracy: 0.7770 - val_loss: 0.4961 - val_accuracy: 0.8235
```

Figure 8. CNN training process

After going through the training process, an evaluation is conducted to look for accuracy, precision, recall, f1-score, specificity, and error rate. From the scenarios used, namely 16 scenarios, the resulting models from 50 epochs to 300 epochs are a total of 96 models. Table 3 shows the average of 10-fold cross-validation for each scenario and epoch.

Table 3. CNN Average Accuracy Testing Results for Every Epoch

Scenario	50 Epoch	100 Epoch	150 Epoch	200 Epoch	250 Epoch	300 Epoch
1	81.287%	82.5%	81.765%	82.5%	81.801%	84.301%
2	79.449%	80.037%	81.765%	83.529%	83.051%	84.118%
3	80.037%	84.228%	81.103%	82.978%	83.088%	84.963%
4	80.478%	81.176%	84.301%	81.765%	83.75%	85.368%
5	79.375%	80.588%	81.765%	81.875%	84.265%	85.993%
6	79.044%	80.515%	80.699%	81.36%	83.125%	80.772%
7	80.221%	78.897%	81.875%	83.603%	82.941%	80.551%
8	83.051%	81.213%	78.824%	82.39%	82.537%	84.191%
9	82.5%	84.779%	84.338%	85.294%	87.206%	88.529%
10	83.64%	84.118%	85.515%	86.213%	86.728%	88.529%
11	84.963%	83.456%	84.191%	86.691%	84.412%	84.779%
12	83.787%	85.368%	82.941%	86.654%	88.603%	88.493%
13	84.412%	85.551%	84.743%	85.441%	84.338%	**89.743%**
14	80.588%	83.75%	82.574%	86.581%	85.919%	85.368%
15	86.213%	83.088%	84.926%	84.485%	84.338%	87.868%
16	82.426%	83.64%	84.706%	86.728%	85.441%	86.654%
Average	**81.967%**	**82.682%**	**82.877%**	**84.256%**	**84.472%**	**85.639%**

Table 3 shows that the more the number of epochs is increased during training, the more accuracy will increase. This is shown in the average of each epoch. However, this can also affect the training time because the more epochs, the longer the training should be. The highest accuracy results reach 89.743% in scenario 13, epoch 300, using the main architecture, namely three convolution layers. This shows that more layers give better accuracy results. The best hyperparameters and regularization for this scenario are using a batch size of 32, regularization of 0.0001, and dropout of 0.2. In this scenario, larger batch sizes produce higher accuracy compared to smaller sizes. This is because a larger batch size will speed up the network computing process [[23]]. Smaller regularization and dropout sizes produce higher accuracy. Smaller regularization indicates that the regulation is not too strong, so the model tends to focus more on the training data and can achieve better accuracy. Then, a smaller dropout allows the model to retain more information compared to a higher dropout value, thereby preventing greater overfitting. So, scenario 13 combines hyperparameter measures and optimal regularization techniques in detecting pneumonia. Table 4 displays the accuracy of the best-performing scenarios (scenario 13) with precision, recall, f1-score, specificity, and error rate.

Table 4. Test Evaluation Results in Scenario 13 for 10 Fold

Fold	Accuracy	Precision	Recall	F1-Score	Specificity	Error Rate
1	94.118%	83.333%	100%	90.909%	100%	5.882%
2	76.471%	77.778%	77.778%	77.778%	75%	23.529%
3	88.235%	90.909%	90.909%	90.909%	83.333%	11.765%
4	94.118%	100%	91.667%	95.652%	83.333%	5.882%
5	88.235%	100%	77.778%	87.5%	80%	11.765%
6	100%	100%	100%	100%	100%	0%
7	100%	100%	100%	100%	100%	0%
8	81.25%	88.889%	80%	84.211%	71.429%	18.75%
9	87.5%	100%	75%	85.714%	80%	12.5%
10	87.5%	100%	71.429%	83.333%	81.818%	12.5%
Average	89.743%	94.091%	86.456%	89.601%	85.491%	10.257%

Table 4 shows the evaluation of the model for each fold. Accuracy, on average, produces good results. Based on the precision-recall values, the CNN model shows a balance in recognizing both pneumonia and normal images. In addition, the model also produces a good F1 score and specificity, as well as a small error rate.

From the previous training process, it can be observed through the accuracy and loss graphs for training and validation in Figure 9. This graph can also show the model experiencing good fitting, underfitting, or overfitting. If the line between training and validation (test) follows each other from the zero point up to the highest accuracy, the training process is said to fit well. However, if the training and validation lines move away from each other as the epoch progresses, then the training process is overfitting/underfitting. Likewise, the loss will be considered good fitting if the training and validation lines follow each other from the highest to the lowest. The following is one of the results.

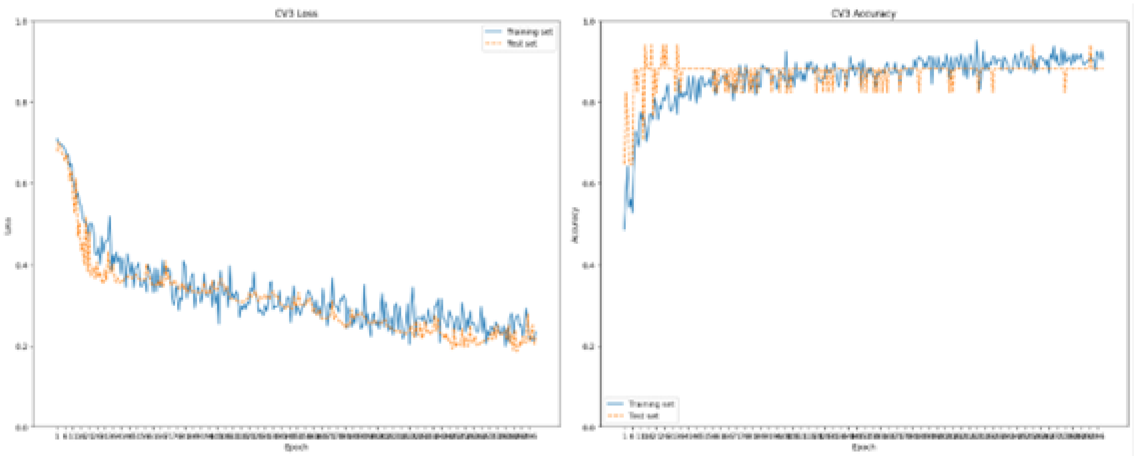


Figure 9. Accuracy and loss training and validation

Figure 9 shows that the training and validation lines for accuracy increase as the epoch progresses. The training and validation lines at a loss decrease as the epoch progresses. This shows no overfitting or underfitting because the two lines, training, and validation, are not far apart.

The discussion is based on previous research by [15], which also uses deep learning to detect pneumonia. This study uses simple CNN with data augmentation and simple CNN without data augmentation. In this study, 10 layers of CNN are used. Then, research conducted by [3] used 6 convolution layers with data augmentation. Table 5 is a comparison of previous research with this research.

Table 5. Comparison of Previous Research and Proposed Research

Research	Method	Accuracy
Research by [15]	Simple CNN without Data Augmentation	0.72%
	Simple CNN with Data Augmentation	0.78%
Research by [3]	CNN with Data Augmentation	77%
Proposed research	CNN with data augmentation	89.743%

Based on the comparison results shown in Table 5, this research, which uses the CNN method with a three-layer convolution architecture or main architecture, provides more optimal accuracy compared to using 6 layers and 10 layers from previous research, which used CNN alone. Apart from that, this shows that simpler layers provide better results than more layers because the more complex the model, the more complex the model tends to be overfitting, and the more layers, the more computation.

4. CONCLUSION

From the comparison of the two architectural models, it was found that the architecture with three convolution layers with data augmentation gave better results than the architecture with two convolution layers with data augmentation. The best model was obtained with hyperparameter settings and regularization techniques, such as epoch 300, batch size 32, L2 regularization 0.0001, and 0.2 dropouts. The larger the epoch, the better the average accuracy. The best accuracy produced was 89.743%. The model is said to be quite good because it can also recognize both pneumonia and normal images with balanced recall, precision, and specificity figures. The precision test results reached 94.091%, recall 86.456%, f1-score 89.601%, and specificity 85.491%. So, the model built in this study has the potential for detecting diseases using image data. In addition, further research is hoped to improve the model's accuracy using different architectures, parameters with different sizes, and more epochs. Furthermore, this research has a limited dataset, so it is hoped that further research can add more datasets so that the model can learn even better.

5. ACKNOWLEDGEMENTS

We thank INSTIKI for the support, both morally and financially, for the successful implementation of this research. We also express our deepest gratitude to the MATRIK journal for agreeing to publish the results of our research. Hopefully, this research can be useful for future research.

6. DECLARATIONS

AUTHOR CONTRIBUTION

We thank author 1 for the formulation of the problem and research objectives and the analysis carried out on this research work. We thank authors 2 and 3 for constructing the CNN model and all the experiments in this study. We thank author 4 for providing the data needed in this study.

FUNDING STATEMENT

Longer training with the support of a higher number of epochs improves the quality of the CNN model in recognizing images. The CNN model with three convolution layers, batch size 32, L2 regularization 0.0001, and dropout 0.2 can recognize a total of 165 lung images with an accuracy of 89.743%. The model built with this architecture can also recognize both classification classes in a balanced and good manner.

COMPETING INTEREST

This research is an annual research grant scheme for INSTIKI lecturers, which is funded by the campus in June 2023.

REFERENCES

[1] V. N. Sutriana, M. N. Sitaresmi, and A. Wahab, "Risk factors for childhood pneumonia: a case-control study in a high prevalence area in Indonesia," *Clinical and Experimental Pediatrics*, vol. 64, no. 11, p. 588, 2021.

[2] M. K. Prastika and E. Astutik, "The Relationship Between Malnutrition and Severe Pneumonia Among Toddlers in East Java,

Indonesia : An Ecological Study," *Journal of Public Health Research and Community Health Development*, vol. 6, no. 2, pp. 93–101, mar 2023.

[3] F. Chandra, S. Arisgraha, R. Rulaningtyas, and M. A. Kusumawardani, "Classification of Pneumonia from Chest X-ray Images Using Keras Module TensorFlow," *Indonesian Applied Physics Letters*, vol. 4, no. 1, pp. 38–44, aug 2023.

[4] I. G. A. A. D. Indradewi, N. W. S. Saraswati, and N. W. Wardani, "COVID-19 Chest X-Ray Detection Performance Through Variations of Wavelets Basis Function," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 1, pp. 31–42, nov 2021.

[5] R. a. Rizal, I. S. Girsang, and S. A. Prasetiyo, "Klasifikasi Wajah Menggunakan Support Vector Machine (SVM)," *REMIK: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 3, no. 2, pp. 1–4, mar 2019.

[6] N. W. S. Saraswati, N. W. Wardani, and I. G. A. A. D. Indradewi, "Detection of Covid Chest X-Ray using Wavelet and Support Vector Machines," *International Journal of Engineering and Emerging Technology*, vol. 5, no. 2, pp. 116–121, dec 2020.

[7] R. Sujatha, J. M. Chatterjee, N. Z. Jhanjhi, and S. N. Brohi, "Performance of deep learning vs machine learning in plant leaf disease detection," *Microprocessors and Microsystems*, vol. 80, p. 103615, feb 2021.

[8] M. F. Naufal, "Analisis Perbandingan Algoritma SVM, KNN, dan CNN untuk Klasifikasi Citra Cuaca," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 2, pp. 311–318, mar 2021.

[9] A. Peryanto, A. Yudhana, and R. Umar, "Convolutional Neural Network and Support Vector Machine in Classification of Flower Images," *Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika*, vol. 8, no. 1, pp. 1–7, mar 2022.

[10] L. Zhou, C. Zhang, F. Liu, Z. Qiu, and Y. He, "Application of Deep Learning in Food: A Review," *Comprehensive Reviews in Food Science and Food Safety*, vol. 18, no. 6, pp. 1793–1811, nov 2019.

[11] T. A. Zuraiyah, S. Maryana, and A. Kohar, "Automatic Door Access Model Based on Face Recognition using Convolutional Neural Network," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 22, no. 1, pp. 241–252, nov 2022.

[12] R. A. Ramadhani, B. W. Pangestu, and R. Purbaningtyas, "Klasifikasi Tumor Otak Menggunakan Convolutional Neural Network dengan Arsitektur Efficientnet-B3," *JUST IT : Jurnal Sistem Informasi, Teknologi Informasi dan Komputer*, vol. 12, no. 3, pp. 55–59, aug 2022.

[13] N. Yudistira, A. W. Widodo, B. Rahayudi, and P. Korespondensi, "Deteksi Covid-19 pada Citra Sinar-X Dada Menggunakan Deep Learning yang Efisien," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 6, pp. 1289–1296, dec 2020.

[14] A. Peryanto, A. Yudhana, and R. Umar, "Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation," *Journal of Applied Informatics and Computing*, vol. 4, no. 1, pp. 45–51, may 2020.

[15] I. S. Walia, M. Srivastava, D. Kumar, M. Rani, P. Muthreja, and G. Mohadikar, "Pneumonia Detection using Depth-Wise Convolutional Neural Network (DW-CNN)," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. "6", no. 23, pp. 1–10, sep 2020.

[16] N. Falah, "Rontgen Toraks Normal Tidak Dapat Menyingkirkan COVID-19."

[17] J. P. Cohen, P. Morrison, L. Dao, K. Roth, T. Q. Duong, and M. Ghassemi, "COVID-19 Image Data Collection: Prospective Predictions Are the Future," pp. 1–38, 2020.

[18] V. Shanmukh, "Image Classification Using Machine Learning-Support Vector Machine(SVM)," mar 2021.

[19] H. Darmanto, "Pengenalan Spesies Ikan Berdasarkan Kontur Otolith Menggunakan Convolutional Neural Network," *Joined Journal (Journal of Informatics Education)*, vol. 2, no. 1, pp. 41–59, jul 2019.

[20] A. R. Syulistyo, D. S. Hormansyah, and P. Y. Saputra, "SIBI (Sistem Isyarat Bahasa Indonesia) Translation using Convolutional Neural Network (CNN)," *IOP Conference Series: Materials Science and Engineering*, vol. 732, no. 1, p. 012082, jan 2020.

[21] M. R. R. Allaam and A. T. Wibowo, "Klasifikasi Genus Tanaman Anggrek Menggunakan Metode Convolutional Neural Network (CNN)," *eProceedings of Engineering*, vol. 8, no. 2, apr 2021.

[22] R. R. R. Arisandi, B. Warsito, and A. R. Hakim, "Aplikasi Naïve Bayes Classifier (NBC) pada Klasifikasi Status Gizi Balita Stunting dengan Pengujian K-Fold Cross Validation," *Jurnal Gaussian*, vol. 11, no. 1, pp. 130–139, may 2022.

[23] I. A. Usmani, M. T. Qadri, R. Zia, F. S. Alrayes, O. Saidani, and K. Dashtipour, "Interactive Effect of Learning Rate and Batch Size to Implement Transfer Learning for Brain Tumor Classification," *Electronics 2023, Vol. 12, Page 964*, vol. 12, no. 4, p. 964, feb 2023.