Convolutional Neural Network for Automatic Pneumonia Detection in Chest Radiography

Septy Aminatul Khoiriyah
Informatics and Computer Eng. Dept.
Politeknik Elektronika Negeri Surabaya
Surabaya, Indonesia
septy@it.student.pens.ac.id

Arif Basofi
Informatics and Computer Eng. Dept.
Politeknik Elektronika Negeri Surabaya
Surabaya, Indonesia
ariv@pens.ac.id

Arna Fariza
Informatics and Computer Eng. Dept.
Politeknik Elektronika Negeri Surabaya
Surabaya, Indonesia
arna@pens.ac.id

***Abstract*— X-ray imagery is a non-invasive method that involves exposure to small doses of ionizing radiation to parts of the body to help doctors diagnose diseases, including pneumonia. Detecting pneumonia on a chest X-ray image can be difficult for radiologists because X-ray images are often unclear, overlap with other diagnoses, and approach many other abnormalities. The automated method was developed as a decision support tool to help doctors diagnose pneumonia. This paper proposes different deep convolution neural network architectures with an augmentation strategy to classify the pneumonia detection from the chest X-ray images. We use three convolution layers and three classification layers (fully connected). Resize, flip, and rotation augmentation strategy to avoid overfitting. The experiment result shows that the augmentation strategy on the proposed CNN's architecture results in an accuracy value of 83,38% while on without augmentation result accuracy value 80,25%. The small difference between prediction results with the augmentation strategy and without the augmentation strategy shows that the proposed CNN's architecture can train small datasets.**

Keywords—pneumonia, chest radiography, residual network, convolution neural network

I. Introduction

Pneumonia is one of the lung diseases caused by bacteria, viruses, fungi which resulted in lung inflammation. In general, caused by a variety of microorganisms, such as bacteria, viruses, parasites, fungal chemical exposures, or even physical damage from the lungs from cigarettes or other pollution. Pneumonia or pneumonia that often occurs can be serious, even one that can cause death is community pneumonia.

The Republic of Indonesia's Basic Health Research Data (Riskesdas) in 2018 showed an increase in prevalence, or the number of pneumonia sufferers compared to 2013. Based on the diagnosis of health workers the number of people experiencing this disease disorder in 2018 was around 2 percent, while in 2013 it was 1.8 percent.

The incidence of pneumonia is more common in developing countries, one of which is Indonesia. In 2010 in Indonesia, pneumonia was included in the top 10 hospitalizations in hospitals. The mortality rate of certain diseases or crude fatality rate (CFR) due to this disease over a certain period divided by the number of cases is 7.6 percent. According to the Indonesian Health Profile, pneumonia caused 15 percent of under-five deaths, namely around 922,000 under-five children in 2015. From 2015-2018 confirmed pneumonia cases in children under 5 years increased by around 500,000 per year. Noted the number of patients with pneumonia reached 505,331 patients with 425 patients died.

X-ray imagery is a non-invasive method that involves exposure to small doses of ionizing radiation to parts of the body to help doctors diagnose diseases such as pneumonia, emphysema, lung cancer, line, and tube placement and tuberculosis. The occurrence of pneumonia infection called pneumonia includes inflammation of the lung parenchyma and abnormal alveolar fluid filling. The white portion of the X-ray image shows the lungs are filled with pneumonia-causing fluid, while the black portions represent normal lungs [1]. Detecting pneumonia on a chest X-ray image can be difficult for radiologists because X-ray images are often unclear, overlap with other diagnoses, and approach many other abnormalities. This difference leads to a considerable difference in the diagnosis of pneumonia among radiologists [2]. The automated method was developed as a decision support tool to help doctors diagnose pneumonia [3].

Some researchers detect pneumonia automatically using a digital image processing approach. Image analysis aims to extract useful information by extracting features consisting of segmentation and transformation functions [4]. The process of dividing a digital image into several segments is called segmentation, while the provision of features based on spatial frequency information is called the transformation process. Parveen and Sathik [4] used the Discrete Wavelet Transform (DWT) feature extraction method, Wavelet Frame Transform (WFT), Wavelet Packet Transform (WPT) and detection of pneumonia infection with unsupervised fuzzy c-means classification. Sharma et al. [5] used Otsu thresholding to separate the parts of the healthy lung and those infected with pneumonia in the chest X-ray image. They calculate the ratio of a healthy lung area to the total lung area to detect pneumonia. Ali et al. [6] proposed a method of a cellular neural network to detect symptoms of pneumonia. The cellular neural network uses a 3×3 linear invariant space template based on state equations, output equations, boundary equations, and initial values. The difference in grayscale color on the CT image is the basis for segmentation between normal and pneumonia areas showing good performance results

Recently, deep learning algorithms, particularly the convolution neural networks (CNN), become the leader for the classification of medical images. This is because, in traditional machine learning methods, the selection of feature extraction is time-consuming and varies according to different objects [7]. Research to detect pneumonia has been carried out by several researchers, including Rajpurkar et al

978-1-7281-9530-8/20/$31.00 ©2020 IEEE

[8], Siddiqi [9], and Sharma et al [10] using CNN with different architectures and strategies based on X-ray images. CNN-based learning requires large training data to get superior results. Medical images are difficult to collect because the collection and labeling of medical data are faced with time-consuming privacy requirements and expert explanations. Wang and Perez [11] found that transformation-based data augmentation effectively classifies the image. Image augmentation techniques can avoid overfitting in the training process so that the model obtained is more optimal.

This paper proposes different deep convolution neural network architectures with an augmentation strategy to classify the pneumonia detection from the chest X-ray images. We use three convolution layers and three classification layers (fully connected). Resize, flip, and rotation augmentation strategy to avoid overfitting. The proposed CNN's is evaluated using both original dataset and augmented dataset.

II. Related Work

In the medical imaging domain, the deep convolutional neural network (CNN) approach has become the leading machine learning tools including detection chest radiography as a computer-aided [12]. Traditional pattern recognition systems require engineering to design appropriate feature extraction to convert the pixel value data of an image into a feature vector which is then followed by image classification tasks. Whereas the CNN approach can extract discriminatory features at various levels of abstraction while at the same time doing classification tasks [13].

Siddiqi [11] automates the detection of pneumonia using a sequential 18-layer neural convolutional network model of a publicly available chest X-ray image dataset. This model performs classification tasks with a classification accuracy of 0.9439 and high sensitivity (0.99) but produces lower specificity than expected (0.86).

Rajpurkar [12] developed the CheXNet algorithm, a 121-layer CNN network that was trained on the publicly available ChestX-ray14 dataset capable of detecting 14 diseases with state-of-the-art results.

Sharma et al [13] developed CNN which consisted of convolution layers, max-pooling, and classification layers. The CNN architecture is evaluated with a dropout layer and without a dropout layer. X-ray images that enter CNN are 64 x 64 in size and use augmentation strategies such as rotation, flip, resize, etc. The results show that CNN's performance with dropouts outperformed other models.

III. Methodology

The methodology used in this study can be seen in Figure 1. There are two processes, namely training and testing. The training dataset was augmented to develop the dataset to avoid overfitting. The augmentation strategy consists of resizing (cropping), flip, and rotation. The training dataset is divided into training datasets and validation randomly. The validation dataset is used to provide a model evaluation that does not favor the training dataset. The resulting model will be used for the testing process (prediction). Before the prediction process is carried out, the data testing is pre-processed. Pre-processing of the testing dataset consists of the centered crop, flip, and rotation.

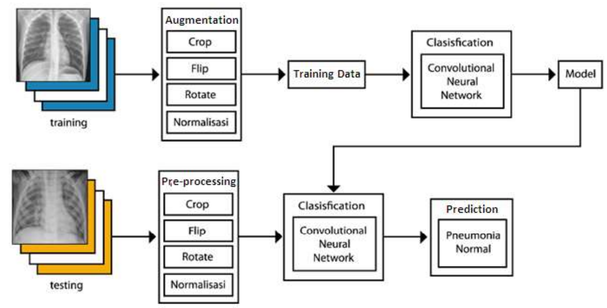

Fig. 1. Metodologi of pneumonia detection using CNN from chest X-ray.

A. Radiography Image Dataset

The dataset used in this paper uses 5,856 X-ray radiographic images of Kaggle [14] in JPEG format with dimensions of 1024×1024 pixels and grouped into two classes: NORMAL and PNEUMONIA, which is stored into a different folder. The dataset is divided into training and testing data with a composition of 90% and 10% randomly. Composition of training and testing dataset can be seen in Table 1. 10% of training dataset are randomly selected as validation data to provide the skills of the model sought by comparing and selecting suitable models.

TABLE I. Compositon of Dataset Training and Testing

Dataset	Pneumonia	Normal	Total
Training	3,883	1,349	5,232
Testing	390	234	624
Total	4,273	1,583	5,856

B. Augmentation

The dataset used in this paper is 5,856 images which are still very small compared to the ImageNet general image dataset which reaches 14,197,122 [9]. To improve the performance of the model used augmentation strategies. 26,160 training dataset generated in the augmentation process.

This study uses a simple transformation augmentation strategy consisting of flipping horizontally with a percentage of 0.5, rotating randomly in the range of -45° to +45°, and cropping randomly to produce an image size of 224×224. Before the training process is carried out, normalization of the dataset is based on the mean and standard deviation.

C. Model Architecture and Training

The convolutional layer is the core building of CNN. Image input passes through a convolutional layer which consists of a set of filters with certain parameters. The filter consists of a height and weight that is smaller than the input size. Filters are built in 3x3 size to ensure a detailed feature reading. Each filter will calculate the ReLu activation value and be pooled to summarize the convolution result information. The output from the convolution process is used as input in the classification process by first flattening. The architecture can be seen in Figure 2.

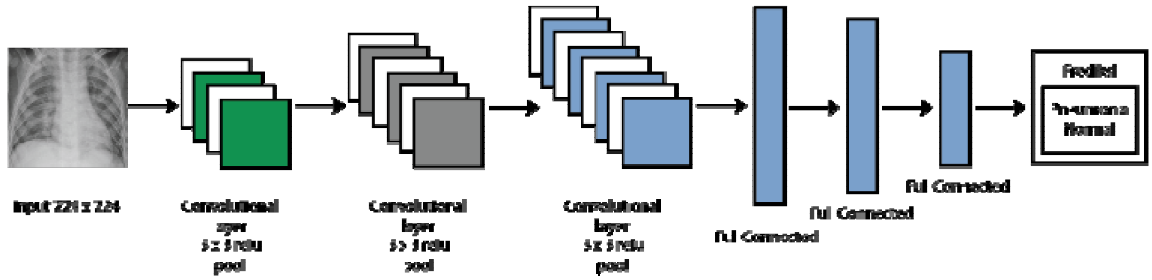

Fig. 2. Proposed CNN's architecture.

Input images that enter CNN are 224×224. The filter used is 3×3 in size to guarantee the learning process can run in detail, followed by a rectified linear unit (ReLU) as a function of activation and pooling of the MaxPool 2×2 layer. Summary of proposed CNN architecture can be seen in Figure 3.

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1          [-1, 8, 220, 220]             608
         MaxPool2d-2          [-1, 8, 110, 110]               0
            Conv2d-3         [-1, 16, 106, 106]           3,216
         MaxPool2d-4           [-1, 16, 53, 53]               0
            Conv2d-5           [-1, 32, 49, 49]          12,832
         MaxPool2d-6           [-1, 32, 24, 24]               0
            Linear-7                 [-1, 1000]      18,433,000
            Linear-8                  [-1, 500]         500,500
            Linear-9                    [-1, 2]           1,002
================================================================
Total params: 18,951,158
Trainable params: 18,951,158
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.57
Forward/backward pass size (MB): 6.15
Params size (MB): 72.29
Estimated Total Size (MB): 79.01
----------------------------------------------------------------
```

Fig. 3. Summary of proposed CNN's architecture.

In Figure 3, the 224×224 image input after the first convolution will produce an image of 110×110 blocks of 8 blocks, after the second convolution will produce a 49×49 image of 16 blocks, the third convolution produces a 32×24 image of 32 blocks. After that, flattening from 2D image to 1D before the fully connected process. There are 3 layers fully connected (3 hidden layers) to produce 2 classes, namely PNEUMONIA and NORMAL.

The parameter of the training process can be seen in Table II. The CNN model is trained for 100 epoch using Adam optimizer with weight decay 1e-5. The Adam Optimizer [15] demonstrated excellent training speed performance in classification problems. Adam's algorithm is an algorithm capable of adaptive learning in deep learning networks.

During training and validation, we calculate using the cross-entropy loss.The loss function commonly used in CNN traning for image classification problems is cross-entropy. In image classification practice, the cross-entropy loss function functions relatively well in training the training dataset. Cross-entropy predicts the probability distribution $p\ (y = i)$ in each class i = 1,2, ..., C. If the correct class is c, then the cross-entropy loss is

$$\mathcal{L} = -\sum_{i=1}^{C} 1[i=c]\log p(y=i) = -\log p(y=c) \quad (1)$$

TABLE II. TRAINING PARAMETER

Loss	Cross Entropy
Optimizer	Adams
Learning Rate	Static with weight decay 1e-5
Sum of epoch	100

IV. RESULT

This chapter explains the result of system design that has been explained in system design. CNN training uses a GPU with an NVIDIA RTX2070 mini 8GB memory engine specification.

A. *Training Result*

The training process was carried out on a total of 26.1600 image data divided into training data (90%) and validation data (10%). The computational process that occurs to produce train loss is firstly re-declared input and labels from the image in the training dataset. Then use the optimizer.zero_grad function to set the gradients to zero first because in Pytorch the gradient will accumulate in the next looping process so that the gradient is set to zero first and to get the parameters updated correctly.

The next step is to compute the output variable by using the model.forward function to then compute the output tensor from the input tensor. After that, calculate the loss value through the output along with its label by calling the criterion function with parameters in the form of output, the label wherein the criterion function to calculate the loss value. After that, call the loss.backward function to compute the loss gradient with all the parameters contained in the loss which are then stored in the parameter.grad attribute for each parameter. After that the optimizer.step function is called where this function has the function to update all parameters in parameter.grad. Then, calculate the average amount of loss in one iteration of the dataset loader training process using the equation running_loss += loss.item. The calculation of variables in validation loss is also the same as the process for calculating variables in training loss by using the validation dataset.

In convolution neural networks, the loss value will show how good or bad the model is in inputting data and training in a dataset. The smaller the resulting loss value, the better the classifier is in modeling the relationship between input data and target output. Figures 4 and 5 are the plot of decreasing training loss of proposed CNN's architecture with augmentation and without augmentation. Figures 6 and 7 are the plot of increasing training accuracy of proposed CNN's architecture with augmentation and without augmentation.

.

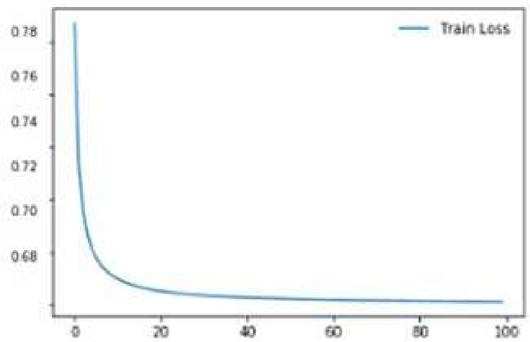

Fig. 4. A plot of training loss decreasing of proposed CNN's architecture with augmentation strategy.

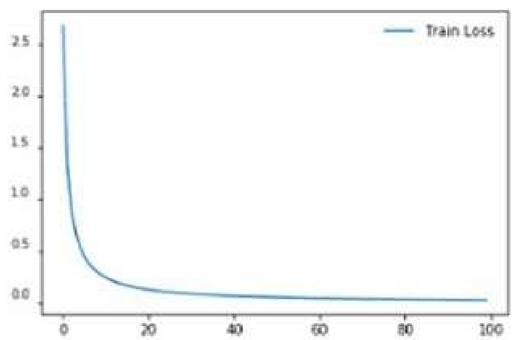

Fig. 5. A plot of training loss decreasing of proposed CNN's architecture without augmentation strategy.

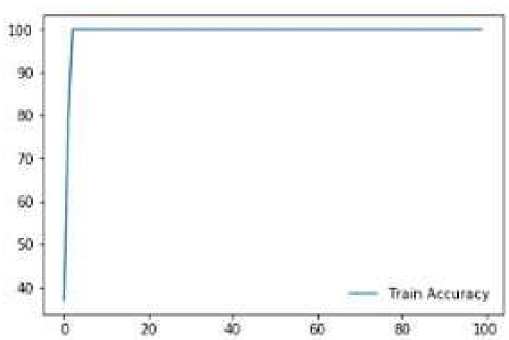

Fig. 6. A plot of training accuracy increasing of proposed CNN's architecture with augmentation strategy.

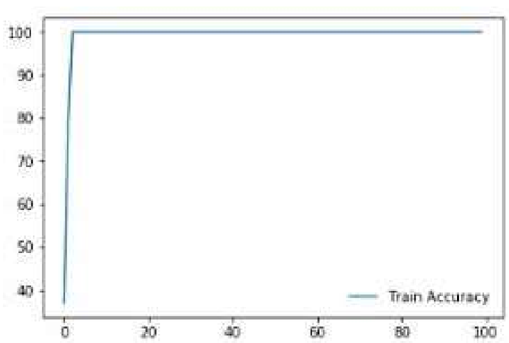

Fig. 7. A plot of training accuracy increasing of proposed CNN's architecture without augmentation strategy.

Table III report the training loss, validation loss, training accuracy, and validation accuracy after 100 epochs of proposed CNN's architecture with augmentation strategy and without augmentation strategy. Figures 4-7 and Table III show that the training process achieves an optimal model of the proposed CNN's architecture. It means that the CNN model and parameter work well to train the training dataset (huge and small dataset).

TABLE III. TRAINING LOSS, VALIDATION LOSS, TRAINING ACCURACY, AND VALIDATION ACCURACY AFTER 100 EPOCHS

Type	Proposed CNN's architecture with augmentation	Proposed CNN's architecture with augmentation
Training Loss	0,6602	0,6943
Validation Loss	0,6423	0,6735
Training accuracy	100,00	100,00
Validation accuracy	100,00	100,00

B. Classification Result

Evaluation of the classification results is used to determine the performance of the proposed CNN's architecture to detect pneumonia. Evaluation is measured using accuracy to evaluate the system process from the average error between pneumonia prediction results and the real value from the labels. The highest value of accuracy, the better result it will be. The accuracy equation is

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Each TP, TN, FP, and FN represent True Positive, True Negative, False Positive and False Negative.

The evaluation compares the performance of the proposed CNN's architecture with augmentation strategy and without augmentation of 624 testing data. The accuracy of the proposed CNN's architecture with the augmentation strategy reached 83.38%, while the accuracy of the proposed CNN's architecture without augmentation strategy reached 80.25%. This result shows that the augmentation strategy on the proposed CNN's architecture achieves better results than without augmentation. In general, the proposed CNN's architecture achieves good predictions (more than 80%). The small difference between prediction results with the augmentation strategy and without the augmentation strategy shows that the proposed CNN's architecture can train small datasets.

Visualization of prediction results on 5 random images using the proposed CNN's architecture with augmentation strategies can be seen in Figure 8. The figure is the appearance after the normalization process.

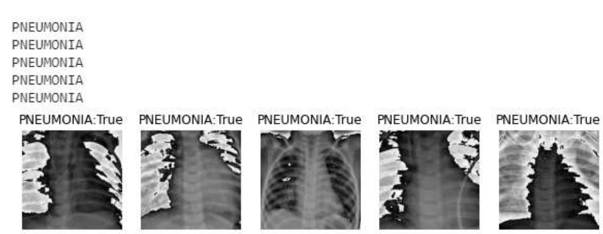

Fig. 8. Five random samples of prediction result.

V. Conclusion

This research proposes CNN's architecture with an augmentation strategy to detect the pneumonia disease using a chest X-ray image. The experiment result shows that the augmentation strategy on the proposed CNN's architecture achieves better results than without augmentation. The small difference between prediction results with the augmentation strategy and without the augmentation strategy shows that the proposed CNN's architecture can train small datasets.

References

[1] http://en.wikipedia.org/wiki/Chestradiograph.

[2] M. I. Neuman, E. Y. Lee, S. Bixby, S. Diperna, J. Hellinger, R. Markowitz, S. S. Servaes, M. C. Monuteaux, S. Shah, “Variability in the interpretation of chest radiographs for the diagnosis of pneumonia in children”, Journal of hospital medicine, Vol. 7 No. 4, pp. 294-298, 2012.

[3] M. Fiszman, W. W. Chapman, D. Aronsky, R. S. Evans, P. J. Haug, “Automatic detection of acute bacterial pneumonia from chest X-ray reports”, Journal of the American Medical Informatics Association, Vol. 7 No. 6, pp. 593-604, 2000.

[4] N. Parveen, M. M. Sathik, “Detection of pneumonia in chest X-ray images”, Journal of X-ray Science and Technology, Vol. 19 No. 4, pp. 423-428, 2011.

[5] A. Sharma, D. Raju, S. Ranjan, “Detection of pneumonia clouds in chest X-ray using image processing approach”, In 2017 Nirma University International Conference on Engineering (NUiCONE), pp. 1-4, 2017, IEEE.

[6] A. A. Abdullah, N. M. Posdzi, Y. Nishio, “Preliminary study of pneumonia symptoms detection method using cellular neural network”,. In International Conference on Electrical, Control and Computer Engineering 2011 (InECCE), pp. 497-500, 2011, IEEE.

[7] D.S. Kermany, M. Goldbaum, W. Cai, C.C. Valentim, H. Liang, S.L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan, “ Identifying medical diagnoses and treatable diseases by image-based deep learning”, Cell, Vol. 172 No.5, pp. 1122–31, 2018

[8] L. Perez, J. Wang, “The effectiveness of data augmentation in image classification using deep learning”, 2017, arXiv preprint arXiv:1712.04621.

[9] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, M.P. Lungren, “Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning”, 2017, arXiv preprint arXiv:1711.05225.

[10] R. Siddiqi, “Automated Pneumonia Diagnosis using a Customized Sequential Convolutional Neural Network”, In Proceedings of the 2019 3rd International Conference on Deep Learning Technologies (pp. 64-70), 2019

[11] H. Sharma, J. S. Jain, P. Bansal, S. Gupta, “Feature Extraction and Classification of Chest X-Ray Images Using CNN to Detect Pneumonia”. In 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2020, pp. 227-231).

[12] C. Qin, D. Yao, Y. Shi, Z. Song, “Computer‐aided detection in chest radiography based on artificial intelligence: a survey”, BioMedical Engineering OnLine, Vol. 17 No. 1, pp. 113, 2018.

[13] H. Greenspan, B. Van Ginneken, R. M. Summers, “Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique”, IEEE Transactions on Medical Imaging, Vol/ 35 No. 5, pp. 1153-1159, 2016.

[14] https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

[15] D.P. Kingma, and J. Ba, “Adam: A method for stochastic optimization”, arXiv preprint arXiv:1412.6980, 2014.