

Classification of pneumonia from X-ray images using siamese convolutional network

Kennard Alcander Prayogo, Alethea Suryadibrata, Julio Christian Young

Department of Informatics, Universitas Multimedia Nusantara, Indonesia

Article Info

Article history:

Received Aug 15, 2019

Revised Jan 20, 2020

Accepted Feb 23, 2020

Keywords:

Chest X-Ray

Image classification

Pneumonia

Siamese Convolutional Network

ABSTRACT

Pneumonia is one of the highest global causes of deaths especially for children under 5 years old. This happened mainly because of the difficulties in identifying the cause of pneumonia. As a result, the treatment given may not be suitable for each pneumonia case. Recent studies have used deep learning approaches to obtain better classification within the cause of pneumonia. In this research, we used siamese convolutional network (SCN) to classify chest x-ray pneumonia image into 3 classes, namely normal conditions, bacterial pneumonia, and viral pneumonia. Siamese convolutional network is a neural network architecture that learns similarity knowledge between pairs of image inputs based on the differences between its features. One of the important benefits of classifying data with SCN is the availability of comparable images that can be used as a reference when determining class. Using SCN, our best model achieved 80.03% accuracy, 79.59% f1 score, and an improved result reasoning by providing the comparable images.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Alethea Suryadibrata,

Department of Informatics,

Universitas Multimedia Nusantara,

Scientia Boulevard St., Gading Serpong, Tangerang, Banten 15811, Indonesia

Email: alethea@umn.ac.id

1. INTRODUCTION

Pneumonia is one of the leading causes of mortality in children under 5 years old besides preterm birth complications [1]. In 2015, pneumonia is responsible for around 15% of all deaths or almost 920,136 number of children in this age group [2]. Pneumonia is an inflammation of lung tissue as a result of infectious agents [3]. The most common type of pneumonia is bacterial pneumonia and viral pneumonia [4]. Bacterial pneumonia is caused by bacteria such as *Streptococcus pneumoniae* *Haemophilus influenzae* type b (Hib), and *Mycoplasma pneumoniae* [5]. Viral pneumonia is often caused by respiratory virus including influenza, parainfluenza virus, and adenovirus [6].

The main problem of pneumonia treatment is the difficulty to make a clinical decision as a result of the inability to identify the infectious organism [7]. To overcome this problem, pneumonia disease is usually treated using antibiotics. However, the use of antibiotics to treat viral pneumonia is ineffective [4] and misuse of antibiotics can increase the risk of antibiotic resistance [7]. Thus, it is important to identify the type of microorganism on the diagnosis of pneumonia to receive suitable treatments. Medical imaging techniques has been used to support diagnosis result such as ultrasonography, radiography, and tomography. Image processing and machine learning algorithms also played important role to assist doctor in making faster and more accurate diagnosis [8]. Some researchers have combined these approaches on medical imaging result i.e. [9] used possibilistic c-means for ultrasound images, [10] used wavelet decomposition for chest radiograph, and [11] used deep neural network for MRI images segmentation.

As there are no test that can fully identify specific cause of pneumonia, a test to distinguish between bacterial and viral pneumonia would be a major advance [12]. The most common test is using host biomarkers (i.e. procalcitonin (PCT), C reactive protein (CRP), white blood cell indicators), but these tests resulting in complex outcome and is lack of an accurate reference comparator test [12]. Accordingly, chest x-ray (CXR), a medical imaging technique, is used to provide more comprehensive clinical signs. In clinical decision, CXR supports bacterial or viral etiology and even for complications [7]. As a result, in this work, deep learning method will be used to classifying the result of CXR into normal condition, bacterial pneumonia, and viral pneumonia.

Deep learning method that mainly used for image problem is convolutional neural network (CNN). Several researchers have used CNN to classify medical imaging results such as [13] for classifying stroke, [14] for classifying type of muscle, and [15] for classifying abdominal ultrasound images. The used of CNN for classifying CXR has also been performed in [16] and [17] for classifying two classes, normal condition and pneumonia. The result shows high accuracy in both works. However, the predicted result still lacks on reasoning part that supports the results.

Another variation of CNN is siamese convolutional network (SCN). SCN using two identical CNN with the same architecture and the same weight [18]. This network receives a pair of images as input and returns a similarity score between them. SCN has been used by [19] on tracking problems and is able to perform well in those problems. Another work is done by [20] for one shot character recognition. They form a pair from a single data image to suit the SCN input. For the one-shot learning problem, they used different class collections of alphabets for testing and classified each of the alphabets using SCN. Whereas our work will use similar type of approach but with the same class of data. Using this approach, we could also use the compared images to support classification results, considering the fact that similar methods still deficient in the reasoning part.

2. RESEARCH METHOD

2.1. Data description

The used dataset is Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification [16]. This dataset contains 5863 chest x-ray (CXR) images from patients one to five years of age at Guangzhou Women and Children's Medical Center [16]. It also divides the data into 3 classes, normal condition, bacterial pneumonia, and viral pneumonia. Diagnoses of all CXR images have also been reviewed by two radiologists [16].

2.2. Proposed model

2.2.1. Siamese convolutional network architecture

Figure 1 described siamese convolutional network (SCN) architecture that will be used in this work. This network received a pair of 224x224 chest x-ray images as input. Then, input will be fed into convolutional network to extract features from each image. Connection function was used for combining each output of convolutional network. We used cosine distance as a connection function to evaluate similarity between each input. Cosine distance has been used for pattern recognition problems as this function is invariant to the magnitudes of samples [20]. This function formula is [21].

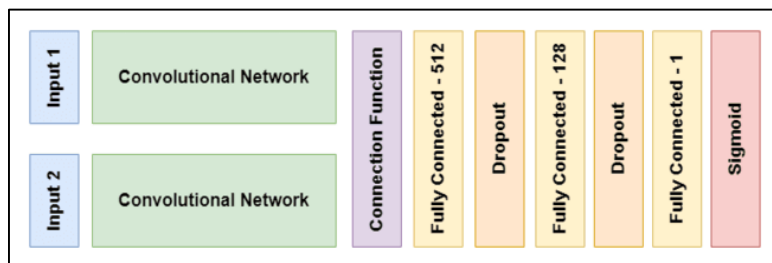


Figure 1. Overall siamese convolutional network architecture

$$\text{cosine distance} = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n a_i b_i}{\sqrt{\sum_{i=1}^n a_i^2} \sqrt{\sum_{i=1}^n b_i^2}} \quad (1)$$

where A and B are the output of convolutional network as a vector with corresponding to $A = a_1, a_2, \dots, a_n$ and $B = b_1, b_2, \dots, b_n$ n will be the number of vector attributes. The output of this connected function will be forwarded into fully connected layer with 2 hidden layers. Dropout layer was added between each hidden layer as a regularization. In the last layer, we used sigmoid activation to produce similarity result that is bound to $[0, 1]$.

Most of the SCN model used parameter sharing on its two convolutional networks weight and bias. According to [22], removing this constraint can deal with more specific matching task than general task. As our problem used the same class for testing, we also did not use parameter sharing in the network. Furthermore, to enhancing matching task for every class, we used separate model for each class. Using this approach, each model learned to gain best features for comparison between corresponding class with other classes.

2.2.2. Convolutional network architecture

In convolutional network, we used 14 layers that contain 9 convolutional layers and 5 max-pooling layers as described in Figure 2. The number of channels from each layer are 32, 64, 64, 64, 64, 128, 128, 128, 128, 256, 256, 256, and 512. Each convolutional layer used a 3x3 filter with two strides and zero paddings. On each convolutional layer, ReLU activations will be used to remove negative value. The max-pooling filter size is 2x2 with two strides and zero paddings. This way, the image will be down sampled and focusing on important features. To reduce overfitting, regularization techniques such as dropout [23] and batch normalization will be used [24]. The output of this network has a size of 1x1x512 and will be flattened into vector 1x512 before forwarded to fully connected network.

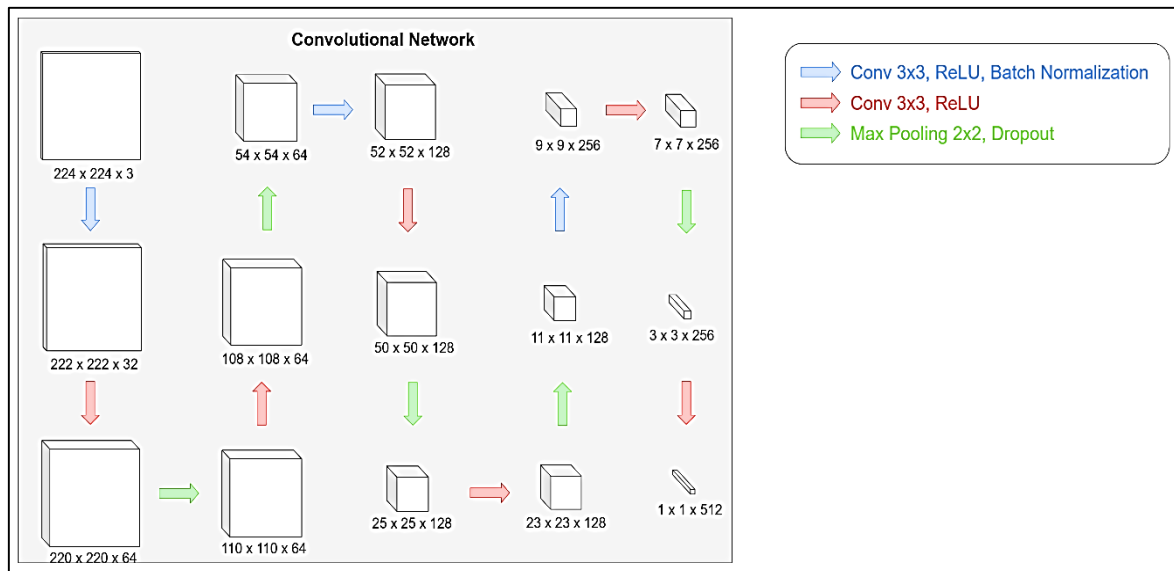


Figure 2. Selected convolutional network for SCN architecture

2.3. Splitting dataset

In this work, we divided the dataset into training (70%), validation (10%), and testing (20%) sets. this training and validation data will be used for pair creation. As a result, we used different amounts of data for training and validating the network.

2.4. Pair creation

As SCN input comprises a pair of images, a preprocessing method to form a pair from the dataset must be performed. Figure 3 shows set of pairs that will be fed into the network. For same class pair or genuine pair, we paired data from the same class. While on the different class pair or impostor pair, the first input always comes from the same class that corresponds to the model. The second input consists of data from different classes than the first input. In our work, we used an equal number of genuine pairs and impostor pairs to avoid class imbalance. The amount of created genuine pairs can be calculated using combinatorics. For n number of data from the same class, there are $\binom{n}{2}$ amount of pair data available to use. Hence, the split data training and validation will be processed into a form of data pair. Using these pair data, we randomly picked 10000 pair of training data and 2000 pair of validation data for training phase.

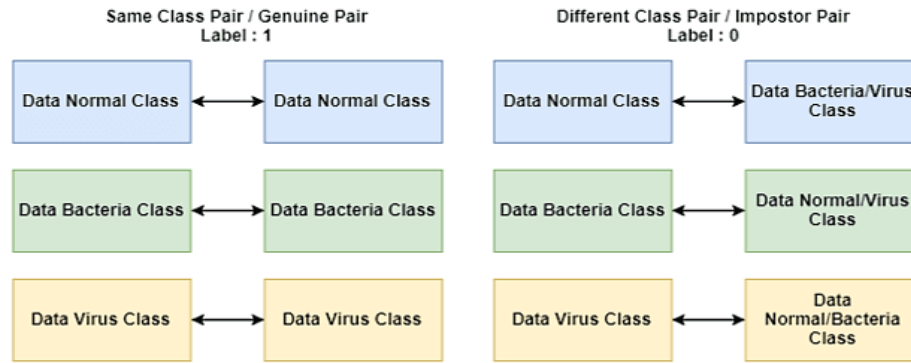


Figure 3. Set of pairs for each class model

2.5. Training

We trained our SCN using mini batch with batch size 16. To generate each batch, the paired data were put into generator. In addition, generator also performed additional preprocessing to the images. First, images were resized into 224x224. Then, flip horizontal augmentation was performed randomly to the images. We also normalized the images based on the mean and standard deviation from 1000 samples training data. Figure 4 shows preprocessing flow in the generator. To train the network, Adam optimizer was used to optimize the network. The chosen hyperparameters are shown in Table 1.

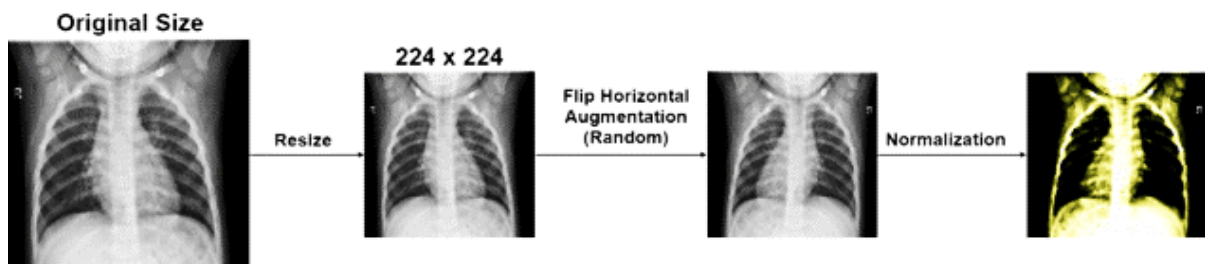


Figure 4. Preprocessing in the generator

Table 1. SCN Hyperparameter Settings

Hyperparameter	Value
Learning Rate	0.001
Dropout Rate	0.5
Learning Decay	0
β_1	0.9
β_2	0.999

As the output of this network is a binary classification, which similarity result was measured, we used binary cross entropy (BCE) for the loss function. BCE is usually used by SCN with sigmoid activations on the last layer [25]. As shown in (2) shows the loss function formula which guide our SCN learning process [26],

$$c = \frac{1}{m} \sum_{k=1}^m [t_k \ln y_k + (1 - t_k) \ln(1 - y_k)] \quad (2)$$

where C is the BCE loss function value from data k to m , t_k and y_k is an actual target and prediction result of data k .

Network from each class was trained for 20 epochs, as the loss has been increasing around the last 5 epochs. The best weights configuration was saved associates to the lowest validation loss. Such weight will be loaded to the network before using it for testing data.

2.6. Testing

Initially, we changed the testing data into a pair. Generator was used to create a pair between testing data with randomly picked data training for each representative class. Figure 5 describes the input class from each pair. The first input always consists of the data corresponding to the class model. For the second input,

we put the testing data, which can be the data from any class. Using these rules, there was no alteration between the paired data input used for training and testing. Thus, the model was able to find similarity features for each class. We defined rank as the number of repeated testing on the same category. This term was also used in [21] to evaluate the network. Then, classification result was based on maximum similarity from C number of classes as shown in 3 [20].

$$C^x = \operatorname{argmax}_c p^{(c)} \quad (3)$$

where C^* defines classification result from all class, c as list of all class, and $p^{(c)}$ as mean prediction result based on number of ranks.

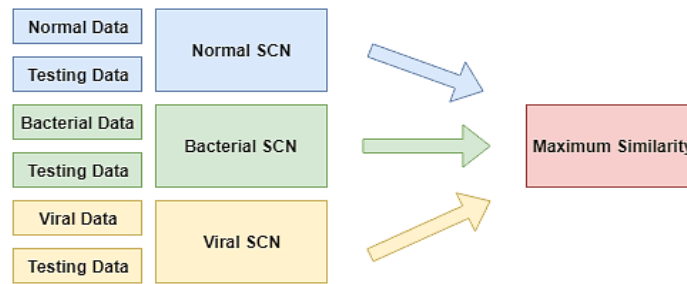


Figure 5. Overall testing flow

3. RESULTS AND ANALYSIS

3.1. Comparing architecture

First, we wanted to compare our architecture with the other ones that have been used for pneumonia classification. The compared network is InceptionV3 nets which are used by [16] and network from [17]. As both architectures are CNN, we drop the fully connected layer on those networks and use them as a convolutional network for our SCN. To know a glimpse of each network performance, we used 2000 training data pair and 400 validation data pair. Table 2 shows the accuracy for each architecture.

Table 2. Comparison of Different Architecture

Architecture	Accuracy
This Paper	77.05%
(Saraiva et al.) [17]	73.46%
InceptionV3 [27]	72.18%

The proposed model in this paper achieves higher accuracy than the other network when used for SCN. If we compare the number of layers on those networks, Saraiva model consist of 9 layers and InceptionV3 has 48 number of layers, whereas our architecture is formed by 14 layers. Therefore, our architecture has deeper layer than Saraiva model, but not as deep as InceptionV3 nets.

3.2. Evaluation

Table 3 details the training results that consist of lowest loss validation, validation accuracy, and epoch position for each class. Based on the loss validation, each model performs quite well to learn similarity between each class. Thus, these models are used to predict testing data. When predicting data, one benefits of using SCN for classification is the availability of other images for comparison. The amount of compared images is equal to the number of ranks. By doing so, high similarity images can be used for case references. Particularly in medical imaging data, the role of these images is essential to support the classification result. Figure 6 shows the illustration of comparison images with its corresponding similarity value.

Table 3. Training result for each class

Class	Lowest Loss Validation	Validation Accuracy	Epoch Position
Normal	0.10897	96.05%	8
Bacterial	0.39441	83.20%	13
Virus	0.43917	83.10%	5

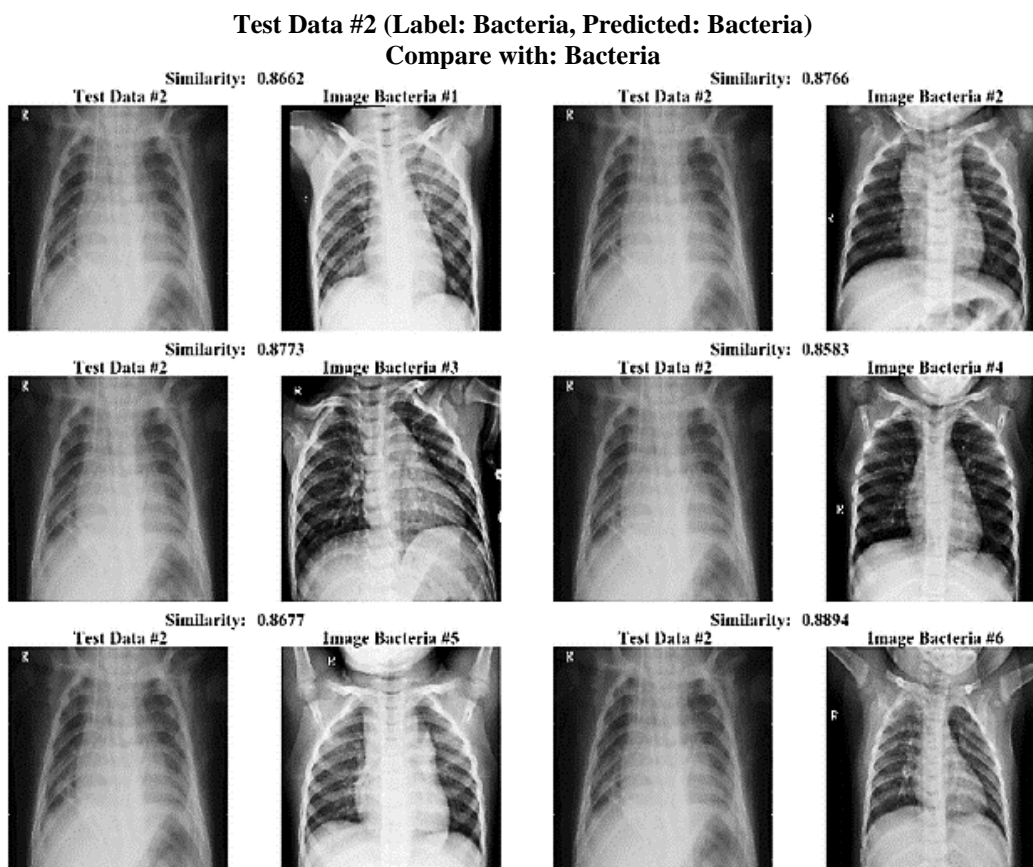


Figure 6. Illustration of comparison visualization for siamese convolutional network

As the prediction result and actual target are also bacteria class, which is a correct classification, each comparison mostly shows high similarity scores. Table 4 describes the result of testing data prediction using different number of ranks. The higher number of ranks, in accordance with higher number of compared data, resulting in higher accuracy. The best accuracy that we achieve is 80.03% using rank-20. Thus, we evaluate this model using confusion matrix, precision, recall, and f1 score. Table 5 shows the confusion matrix for this classification result.

Table 4. Testing result based on number of rank

Rank	Accuracy
1	74.91%
5	77.05%
10	78.33%
20	80.03%

Table 5. Confusion matrix using rank-20 classification

Confusion Matrix		Predicted			Total
		Normal	Bacteria	Virus	
Actual	Normal	297	8	12	317
	Bacteria	15	440	101	556
	Virus	15	83	201	299
	Total	327	531	314	1172

Using this confusion matrix, we calculate other evaluation metrics such as precision, recall, and f1 score. The complete evaluation metrics are described in Table 6. To make sure that our model is not biased to specific class, we compared the overall accuracy with the average f1 score, as this value takes recall and precision into account. If we compared these values, there is no significant difference. However, if we compared f1 score between normal class and virus class, the difference is quite big. When we look at the confusion matrix, most of the wrong classifications on virus class, are classified as bacteria. Intuitively, there are some cases on virus class that are also similar with the bacteria cases. This makes the model prediction was mixed between those two classes. Nevertheless, as there is no significant difference between average f1 score with overall accuracy, our model is not biased to specific class and performs quite well to classify data in general.

Table 6. Complete list of evaluation metrics

Metrics	Normal	Bacteria	Virus	Average
Recall	93.69%	79.13%	67.22%	80.03%
Precision	90.82%	82.86%	64.01%	79.23%
F1 Score	92.24%	80.96%	65.58%	79.59%
Overall Accuracy				80.03%

3.3. Filter visualization

Figure 7 shows filter visualization from the first convolutional layer. In the first input, the extracted features in this layer centering on the right and left side of the lungs. For the bacterial class, the result detailed more on the right side of the lung. On the opposite, the filter result on second input mostly extracts the same features on first convolutional for each class.

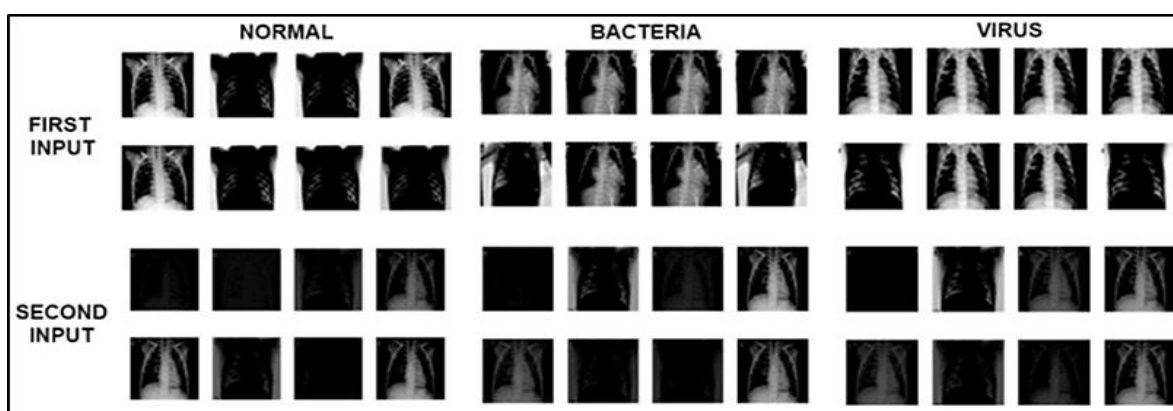


Figure 7. Filter visualization from the first convolutional layer

4. CONCLUSION

In this work, we propose x-ray imaging classification using siamese convolutional network (SCN) which is often used for similarity learning. Our model also drops the constraint of parameter sharing and enhancing features on each class using model separation. The SCN architecture we are used consists of 2 convolutional networks with 14 layers (9 convolutional layers and 5 max pooling layers) respectively, cosine distance as connection function, and fully connected layer with 2 hidden layers. When used for SCN, our architecture achieves better results than other architecture such as inception v3 and model from [17]. This architecture was able to achieve 80.03% accuracy and 79.59% f1 score. We also show the comparison image which used to support the decision from classification result. Therefore, our model has a better result reasoning and details comparing to the similar methods like CNN.

For future works, as the amount of pair result is combinatorics, there is plenty of training data that can be added to improve the model performance. More experiments on hyperparameter settings also help to improve the model. Finally, we plan to improve the result visualization using class activation map (CAM) that is suitable for SCN. It would maximize the purpose of comparison images and give a better understanding of the result.

ACKNOWLEDGEMENTS

This research was supported by the Artificial Intelligence Laboratory in Universitas Multimedia Nusantara.

REFERENCES

- [1] Forum of International Respiratory Societies, "The Global Impact of Respiratory Disease," Second Edition. *Sheffield: European Respiratory Society*, 2017.
- [2] Cilloniz C., Martin-Loeches I., Garcia-Vidal C., San Jose A., Torres A., "Microbial Etiology of Pneumonia: Epidemiology, Diagnosis and Resistance Patterns," *International journal of molecular sciences*, vol. 17, no. 12, December 2016.
- [3] Wojsyk-Banaszak I, Bręborowicz A. Pneumonia in Children. In: Mahboub B. Editors. *Respiratory Disease and Infection*. Rijeka: IntechOpen; 2013: 137-171.

- [4] Tong N. Update on 2004 background paper 6.22 pneumonia, "Priority medicines for Europe and the world A public health approach to innovation," *Geneva, WHO*, 2013.
- [5] Mathur S., Fuchs A., Bielicki J., Van Den Anker J., Sharland M., "Antibiotic use for community-acquired pneumonia in neonates and children: WHO evidence review," *Paediatrics and International Child Health*, vol. 38, pp. S66-S75, November 2018.
- [6] Kim J. E., Kim U. J., Kim H. K., Cho S. K., An J. H., Kang S. J., Park K. H., Jung S. I., Jang H. C., "Predictors of viral pneumonia in patients with community-acquired pneumonia," *PloS one*, vol. 9, no. 12, December 2014.
- [7] Mackenzie, G., "The definition and classification of pneumonia," *Pneumonia*, vol. 8, no. 12, December 2016.
- [8] Razzak M. I., Naz S., Zaib A., "Deep learning for medical image processing: Overview, challenges and the future," *Classification in BioApps*, vol. 26, pp. 323-350, November 2017.
- [9] Suryadibrata A., Kim K. B., "Ganglion cyst region extraction from ultrasound images using possibilistic c-means clustering method," *Journal of Information and Communication Convergence Engineering*, vol. 15, pp. 49-52, 2017.
- [10] Aidoo A. Y., Wilson M., Botchway G. A., "Chest radiograph image enhancement with wavelet decomposition and morphological operations," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol 17, no. 5, pp. 2587-2594, October 2019.
- [11] Al-Kafri A. S., Sudirman S., Hussain A. J., Al-Jumeily, D., Natalia F., Meidia H., Afriliana N., Al-Rashdan W., Bashtawi M., Al-Jumaily M., "Boundary Delineation of MRI Images for Lumbar Spinal Stenosis Detection Through Semantic Segmentation Using Deep Neural Networks," in *IEEE Access*, vol. 7, pp. 43487-43501, 2019.
- [12] Zar H. J., Savvas A., Mark P. N., "Advances in the diagnosis of pneumonia in children," *BMJ*, vol. 358, July 2017.
- [13] Marbun J. T., Andayani, U., "Classification of stroke disease using convolutional neural network," *Journal of Physics: Conference Series*, vol. 978, no. 1, March 2018.
- [14] Katakis S., Barotsis N., Kastaniotis D., Theoharatos C., Tsourounis D., Fotopoulos S., Panagiotopoulos E., "Muscle Type Classification on Ultrasound Imaging Using Deep Convolutional Neural Networks," *2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*, Zagorochoria, pp. 1-5, 2018.
- [15] Cheng P. M., Malh, H. S., "Transfer learning with convolutional neural networks for classification of abdominal ultrasound images," *Journal of digital imaging*, vo. 30, no. 2, pp. 234-243, April 2017.
- [16] Kermany D. S., Goldbaum M., Cai W., Valentim C. C., Liang H., Baxter S. L., McKeown A., Yang G., Wu X., Yan F., Dong J., "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122-1131, February 2018.
- [17] Saraiva A., Ferreira N., Lopes de Sousa L., Carvalho da Costa, N., Sousa, J., Santos D., Valente, A., Soares, S., "Classification of Images of Childhood Pneumonia using Convolutional Neural Networks," in *Proceedings of the 12th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2019)*, pp. 112-119, 2019.
- [18] Barakat B. K., Alasam R., El-Sana J., "Word Spotting Using Convolutional Siamese Network," *2018 13th IAPR International Workshop on Document Analysis Systems (DAS)*, Vienna, pp. 229-234, 2018.
- [19] Leal-Taixé, L., Canton-Ferrer, C., & Schindler, K., "Learning by tracking: Siamese CNN for robust target association," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 33-40, 2016.
- [20] Koch G., Zemel R., Salakhutdinov R., "Siamese neural networks for one-shot image recognition," *ICML deep learning workshop*, Lille, Vol. 2, 2015.
- [21] Rahutomo F., Kitasuka T., Aritsugi M., "Semantic cosine similarity," *The 7th International Student Conference on Advanced Science and Technology ICAST*, Kumamoto, 2012.
- [22] Yi D., Lei Z., Liao S., Li S. Z., "Deep metric learning for person re-identification," *2014 22nd International Conference on Pattern Recognition*, Stockholm, pp. 34-39, 2014.
- [23] Srivastava N., Hinton G., Krizhevsky A., Sutskever I., Salakhutdinov R., "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929-2958, June 2014.
- [24] Ioffe S., Szegedy C., "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *ICML'15: Proceedings of the 32nd International Conference on International Conference on Machine Learning*, vol. 37, pp. 448-456, July 2015.
- [25] Oki H., Miyao J., Kurita T., "Siamese Network for Classification with Optimization of AUC,". In: Gedeon T., Wong K., Lee M. (eds) *Neural Information Processing. ICONIP 2019*. LNCS, Springer, Cham, vol 11954, pp. 315-327, 2019.
- [26] Nasr, G. E., Badr E., Joun C., "Cross Entropy Error Function in Neural Networks: Forecasting Gasoline Demand," *Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference*, pp. 381-384, January 2002.
- [27] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z., "Rethinking the inception architecture for computer vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, pp. 2818-2826, 2016.