

APPLICATION OF LUNG DISEASES DETECTION BASED ON CSLNet

Zulkifli¹, Panji Bintoro^{2,*}, Fitriana³, Sukarni⁴, Abdullah⁵

¹ Informatics Engineering, Faculty of Technology and Informatics, Aisyah University, Indonesia

² Software Engineering, Faculty of Tehcnology and Informatics, Aisyah University, Indonesia

^{3,4} Midwifery, Faculty of Health, Aisyah University, Indonesia

⁵ Nutrition, Faculty of Health, Aisyah University, Indonesia

email: zulkifli@aisyahuniversity.ac.id¹, panjibintoro09@aisyahuniversity.ac.id², fitriana@aisyahuniversity.ac.id³,
sukarni@aisyahuniversity.ac.id⁴, abdullah@aisyahuniversity.ac.id⁵

Abstract

Lung diseases caused by fungal or bacterial infections can lead to inflammation in lung and even death when not detected early. A standard method for diagnosing lung diseases is the use of chest X-ray, which require careful examination of X-ray images by a radiology expert. Therefore, this study proposes several new architecture models, namely CSLNet, to classify chest X-ray images for diagnosing whether patients suffer from COVID-19, viral pneumonia, bacterial pneumonia, tuberculosis, and normal. The experimental results show that the model has an 0.99 average Accuracy, 0.98 Precision, 0.98 Recall, and 0.98 f1-score. Meanwhile, the Receiver Operating Characteristic (ROC) for bacterial pneumonia, COVID-19, normal, tuberculosis, and viral pneumonia are 0.97, 0.99, 0.99, 0.94, and 0.97 respectively. This study is based on a deep learning with a new model, CSLNet, which can work well on the dataset of chest X-ray images used for diagnosing lung diseases.

Keywords : Lung Disease, CSLNet, CNN, LSTM

Received: 06-10-2023 | **Revised:** 26-10-2023 | **Accepted:** 18-11-2023

DOI: <https://doi.org/10.23887/janapati.v12i3.68815>

INTRODUCTION

Diseases affecting human organs, especially lung, are usually caused by smoking, air pollution, or bacterial infections that can attack the respiratory system, leading to serious health issues. In 2019, lung diseases accounted for 212.3 million cases globally with 74.4 million reported deaths [1]. Previous investigations have established that lung diseases cause natural weakness in the respiratory system, requiring patients to be careful in alleviating these issues [2]. The risk of lung diseases such as tuberculosis, pneumonia, and COVID-19 is high, especially in developing and low-income countries, where millions of people face poverty and unhealthy air pollution [3]. Tuberculosis is an infectious diseases that is globally considered a significant source of mortality among the top 10 causes of death [4-5]. Pneumonia caused over 1.1 million hospitalizations and 50,000 deaths in 2010, with the majority of pneumonia-related deaths occurring in patients above 65 years [6]. Meanwhile, COVID-19 is an infectious diseases caused by an unprecedented virus [7]. As of

September 22, 2022, the number of cases has increased by more than 600 million, and deaths have exceeded 6.5 million. Due to its global rapid spread and respiratory infection, COVID-19 has been declared a pandemic by the World Health Organization (WHO) [8].

Chest X-ray are a fundamental indicator for monitoring and examining severe lung diseases such as tuberculosis, pneumonia, and COVID-19 [9]. They can be used to detect airspace opacity, which is a common finding on chest X-ray, indicating severe lung infections [3-10]. Therefore, early detection of lung diseases has become more important and can be improved with machine learning and deep learning. Several studies have used deep learning to detect lung diseases. These include a proposed CNN [11], which is used to accurately detect and classify COVID-19 diseases from normal (healthy) and pneumonia patients. The proposed work is to create a model and tools for lung diseases detection from radiology data patterns using a combination of *Convolutional Neural Network* (CNN) and *Long Short-Term Memory* (LSTM).

The objectives of this study include:

- To classify and accurately detect lung diseases such as tuberculosis, bacterial pneumonia, viral pneumonia, and COVID-19.
- To check the generalization of the method using a larger X-Ray images dataset.
- To achieve higher performance metrics by comparing STN-CNN-LSTM, CNN-LSTM, and CNN combinations.
- To calculate all performance metrics and compare different parameters of the proposed technique with current techniques.
- To create an STN-CNN-LSTM-based model and tools for lung diseases detection that can be used by medical professionals and the public as a reference for determining the type of lung diseases.

A previous study [11] implemented the binary and multiclass classifications using CNN, where the binary classification yielded the best results. Narula and Kumar also presented the CNN-LSTM technique trained on X-Ray data, using deep learning, and showed that this model provided the best results in classifying COVID-19 diseases [12]. The use of transfer learning was investigated for classifying pneumonia in chest X-ray images [13] and the result showed that transfer learning can provide performance benefits from initial training, with a little fine-tuning. Furthermore, three proposed models, namely ResNet50, InceptionV3, and DenseNet121, were separately trained through transfer learning from scratch. The testing results achieved a 4.1% to 52.5% larger Area Under the Curve (AUC) compared to the previously obtained values. A deep learning-based COVID-19 detection system was also proposed by [14] using X-ray images, with a dataset of 76 COVID-19, 4290 pneumonia, and 1583 normal cases images. This scheme achieved an Accuracy of 98.3% for COVID-19 cases. The use of deep learning in medical images processing and analysis is a challenging topic in the field of AI [15]. A study has proposed [16] a CNN model for pneumonia detection, while another [17] proposed vessel extraction from fundus images. An expert system is also proposed for detection of brain tumors on high-resolution brain magnetic resonance images [18]. This study used a specially designed deep learning called SqueezeNet, which was first proposed in a previous investigation [19]. The proposed deep learning for COVID-19 diagnosis was based on the SqueezeNet architecture, due to its smaller structure compared to well-known pre-trained network designs [20,21]. Wang and

Wong [22] proposed a deep-learning model for automated COVID-19 diagnosis and achieved a classification Accuracy of 92.4% in detection 3 classes labeled normal, pneumonia, and COVID-19. Furthermore, Ioannis et al. [23] improved deep learning, which used 224 approved CT images as input to achieve 98.75% and 93.48% Accuracy scores for classes 2 and 3, respectively. Narin et al. [24] used the ResNet50 model to identify COVID-19 from CX images and achieved 98% classification. The best classification score with a structure involving the ResNet50 model was obtained at 95.38%. Pathak et al. [25] used CT images and transfer learning techniques, which were used in a CNN model for automated COVID-19 diagnosis, and achieved high classification performance.

The remainder of this study is organized as follows: Section 2 presents architectural novelties. Section 3 presents the study framework and methods used. Section 4 presents the experiments and results, while the conclusion is presented in Section 5.

Problem Statement

Recently, X-ray dataset collections have become available on the Kaggle repository [26-27]. In this study, the dataset was implemented using deep learning by combining data augmentation, CNN, Spatial Transformer (STN), and LSTM, which was referred to as CNN STN LSTM (CSLNet).

This study applied a new algorithm, CSLNet, to analyze a large dataset of lung diseases for predicting and lung diseases detection in patients. The dataset used was large, making data processing challenging, and also contained a lot of noise. However, there was no sufficient information to predict diseases easily, which made dataset processing a challenging task.

The patients were classified using the CNN deep learning on their X-ray images. The CNN VGG Data STN (VSDNet) [3] can be considered one of the most promising algorithms. Based on the results, the performance of the new CSLNet was compared with the VSDNet method. Therefore, the main contribution of this study was the development of a new CSLNet algorithm that can predict and detect lung diseases in X-Ray images with greater Accuracy than existing methods.

MATERIAL AND METHOD

Figure 1 showed the proposed structure design for this research, where X-ray images of

lung diseases was used as input to classify and

detect lung diseases.

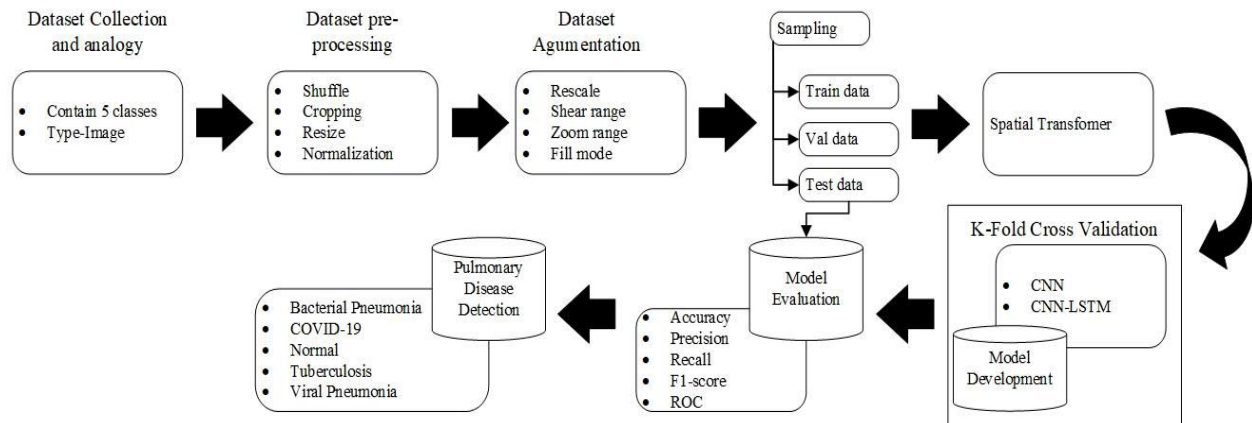


Figure 1. Structural Design of the Research Proposed

Dataset Collection

This study used lung diseases X-Ray dataset [26-27], which currently stored a total of 9039 lung X-Ray images from 5 diseases. The dataset was accessed through the Kaggle

repository and included X-ray images of patients diagnosed with bacterial pneumonia, COVID-19, tuberculosis, viral pneumonia, and normal. The dataset collected was shown in Table 1.

Table 1. Dataset Collection

Data	Number of Images
Bacterial Pneumonia	1301
COVID-19	3677
Normal	1503
Tuberculosis	800
Viral Pneumonia	1758
Total	9039

Dataset Preprocessing and Augmentation

After dataset collection, images preprocessing was performed because:

- Images were taken from different online sources, hence, the dimensions may vary from one source to another.
- Medical images usually contained noise that hindered images quality.

In the process of classification in machine learning or deep learning, the imbalanced class distribution ratio of a dataset can significantly affect the performance of the model [14]. In this study, a dataset of lung diseases was used to train all neural networks tested. As a preprocessing step, X-ray images were shuffled to reduce variance and aid in faster model convergence. The top 5% of chest X-ray images were cropped before training to reduce unnecessary space. At this stage, resizing was

crucial because images were obtained from different online sources with various dimensions varied. The final step was normalization, where the numpy array used to read images was normalized by dividing images matrix by 255. The training of the tested deep neural network architecture required data augmentation with the following configuration, namely rescale (first scale down all chest X-ray images to reduce their size), shear range ($\pm 20^\circ$), zoom range ($\pm 15\%$), and fill mode.

Sampling

In this study, the sample data was partitioned into three parts, namely training (80%), validation (10%), and testing data (10%), as presented in Table 2.

Table 2. Sampling Partition Dataset

Data	Partition	Number of Images	Total
Bacterial Pneumonia	Training (80%)	1041	1301
	Validation (10%)	130	
	Test (10%)	130	
COVID-19	Training (80%)	2943	3677
	Validation (10%)	367	
	Test (10%)	367	
Normal	Training (80%)	1203	1503
	Validation (10%)	150	
	Test (10%)	150	
Tuberculosis	Training (80%)	640	800
	Validation (10%)	80	
	Test (10%)	80	
Viral Pneumonia	Training (80%)	1408	1758
	Validation (10%)	175	
	Test (10%)	175	
Total			9039

Spatial Transformer

CNN represent a very good class of models but are still limited by their inability to become spatially invariant to input data in a

computationally and parameter-efficient manner [28]. STN mechanism is divided into three parts, as shown in Figure 2.

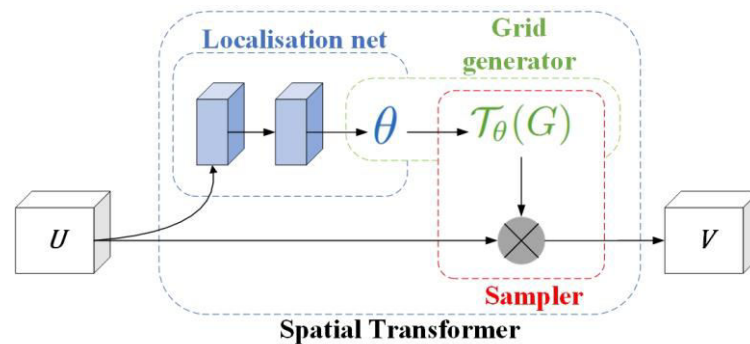


Figure 2. Spatial Transformer of Architecture

Figure 2 shows the input feature map U being passed through a localization network that regresses the transformation parameters θ . The regular spatial grid G over V is transformed to the sampling grid $T_\theta(G)$, which is applied to U , resulting in the output feature map V . The combination of the localization network and sampling mechanism defines STN.

STN mechanism is divided into three parts, and sequentially, the localization network first takes the input feature map. After passing through several hidden layers, it generates the spatial transformation parameters that should be applied to the feature map, resulting in an input-dependent transformation. Subsequently, the predicted transformation parameters are used to create a sampling grid, which is a set of points where the input map is sampled using a network generator to generate the transformed output. Finally, the feature map and sampling grid are

taken as inputs to the sampler, producing an output feature map sampled from the input at the grid points.

Long Short-Term Memory

LSTM is an improvement on Recurrent Neural Networks (RNNs) that proposes memory blocks instead of conventional RNNs units in solving the problem of gradient vanishing and bursting [29]. Furthermore, it incorporates a cell state to store long-term memory, which is the main difference from RNNs. LSTM networks can also remember and connect previous information with the current data [30]. The LSTM is combined with three gates, namely the input, forget, and output gate, where x_t refers to the current input, C_t and C_{t-1} , indicate the current and previous cell state, while h_t and h_{t-1} are the current and previous output, respectively. The internal structure of LSTM is shown in Figure 3.

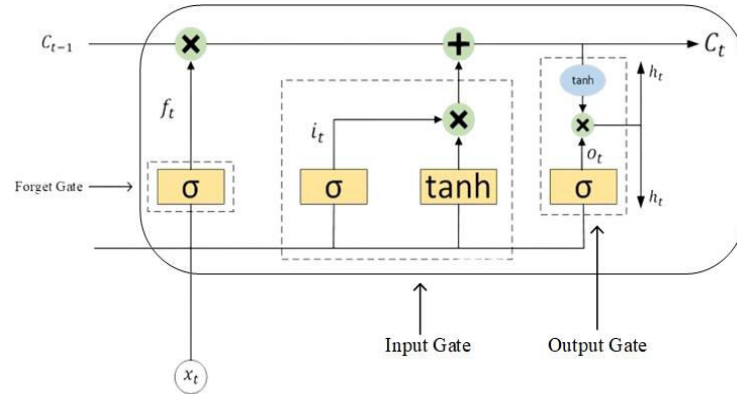


Figure 3. Long Short-Term Memory Structure

The principle of the LSTM input gate is shown in the following formula:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (1)$$

$$\tilde{C}_t = \tanh(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \quad (3)$$

Where (1) is used to pass h_{t-1} and x_t through the sigmoid layer to determine which piece of information to be added. Furthermore, (2) is used to obtain new information after h_{t-1} and x_t are passed through the tanh layer. The current moment information, \tilde{C}_t and long-term memory information C_{t-1} become C_t combined in (3), where W_i refers to the sigmoid output and \tilde{C}_t is the tanh output. Moreover, W_i represents the weight matrix, and B_i is the bias of the LSTM input gate. The LSTM forgetting gate allows selective information paths using the sigmoid layer and the dot product. The decision to forget related information from the previous cell with a certain probability is carried out using (4), where W_f refers to the weight matrix, b_f is the offset, and σ is the sigmoid function.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

The LSTM output gate determines the state required for continuation with inputs h_{t-1} and x_t following (5) and (6). The final output is obtained and multiplied by the state decision vector, which passes the new information, C_t , through the tanh layer.

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = O_t \tanh(C_t) \quad (6)$$

Where W_o and b_o are the weighted matrices of the output gate and the biased LSTM, respectively.

A Combination of CNN and LSTM

Figure 4 shows the proposed model architecture for detection lung diseases, which consists of 5 convolutional blocks, each containing 2 convolutional layers and 1 pooling

layer for blocks 1 to 3. Moreover, blocks 4 and 5 contain 3 convolutional layers and 1 pooling layer, while all blocks have a 20% dropout rate. This makes a total of 12 convolutional and 5 pooling layers used in this study. Kernel size (3,3) is used with 64, 128, 256, 512, and 1024 filters. As the model goes deeper into the layers, the size increases to capture complex features. For activation functions, ReLU (Rectified Linear Unit) is used because this function brings linearity to the CNN. Additionally, in terms of performance, ReLU provides better Accuracy compared to the Sigmoid or Softmax functions [31]. The same padding is used in every convolutional layer, with a dimension step of (1,1).

A pooling layer is used in the CNN architecture to prevent information loss, which occurs every time images shrink after each convolution operation. The max-pooling layer is preferred because it selects the maximum value from the part of images covered by the kernel, which helps extract the most important features from the input matrix. It also suppresses noise from activations along with dimensionality reduction. Kernel size (2,2) is used for max-pooling layers. The features extracted from the CNN layers are fed into a sequence of LSTM layers using the reshape method, with the input size of the LSTM layer being (49, 1024). The 'tanh' activation function is used in each LSTM cell, consisting of a 20% dropout rate, similar to the CNN layers. The proposed architecture summary is shown in Table 2. After analyzing the time characteristics, the architecture sequences X-ray images through fully connected layers to predict whether they belong to one of the 5 categories, namely bacterial pneumonia, COVID-19, viral pneumonia, normal, and tuberculosis.

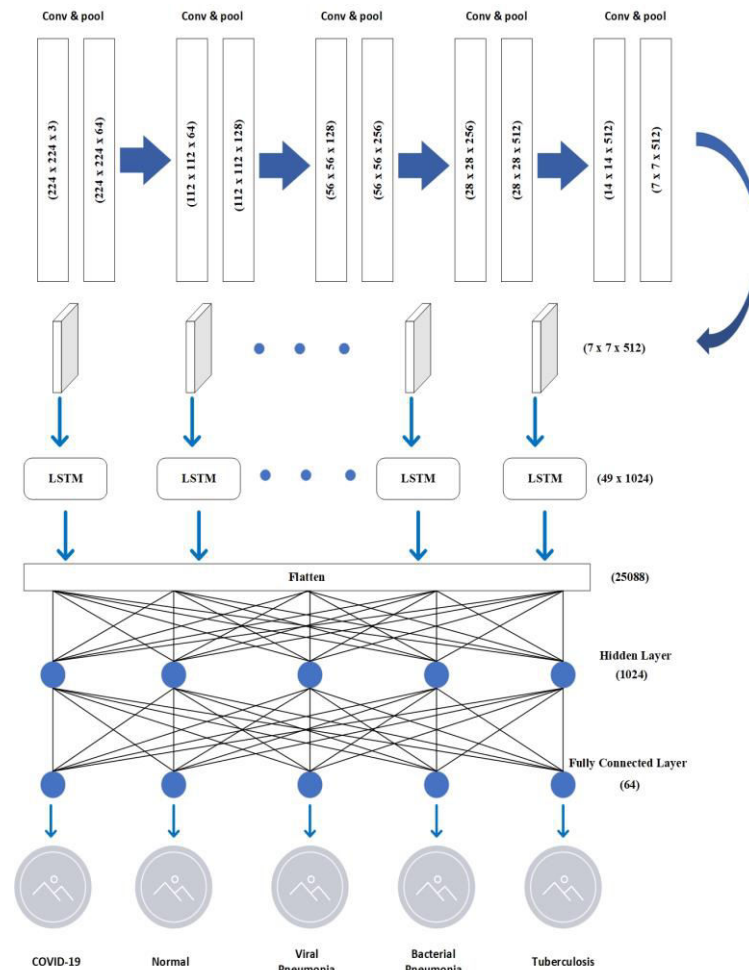


Figure 4. A Combination of CNN and LSTM Networks

Implementation of CSLNet

The algorithms utilized in this study are implemented through the utilization of Jupyter Notebook, Tensorflow, and Keras. The subsequent section provides a detailed description of the implementation process. This represents the fundamental framework of the paper and can be executed on Jupyter Notebook as CSLNet for the entire dataset. The comprehensive architecture of CSLNet is depicted in Figure 5.

The structure contains three key layers in the ensuing order:

- Spatial transformer layers
 - (i) There are five layers.
 - (ii) The first part is lambda λ to transfer the default routing [-0.5:1.0], which indicates that the feature of the lung X-ray images has a normal value of 3.
 - (iii) The second part is batch normalization.
 - (iv) The third part is localization network.
 - (v) The fourth part is grid generator.
 - (vi) The fifth layer is optimal input I .

- Extraction of features layers
 - (i) CNN model has been pre-trained.
 - (ii) CNN architecture has 12 convolutional, 5 pooling layers, and 1 LSTM layer.
- Classification layers
 - (i) In this scenario, the initial layer is referred to as the flattened layer, which is derived from the output of a CNN (Convolutional Neural Network). This CNN output includes five additional classes: 'Bacterial pneumonia', 'COVID-19', 'Normal', 'Tuberculosis', and 'Viral pneumonia'. These extra classes also play a role in determining the sorting, as observed in the simulation. Consequently, they are incorporated into the subsequent layers, forming what is known as the dropout layer.
 - (ii) The final layer consists of dense dropout layers, gradually decreasing in depth.

Figure 5 shows the flowchart of implementation CSLNet for detection lung diseases.

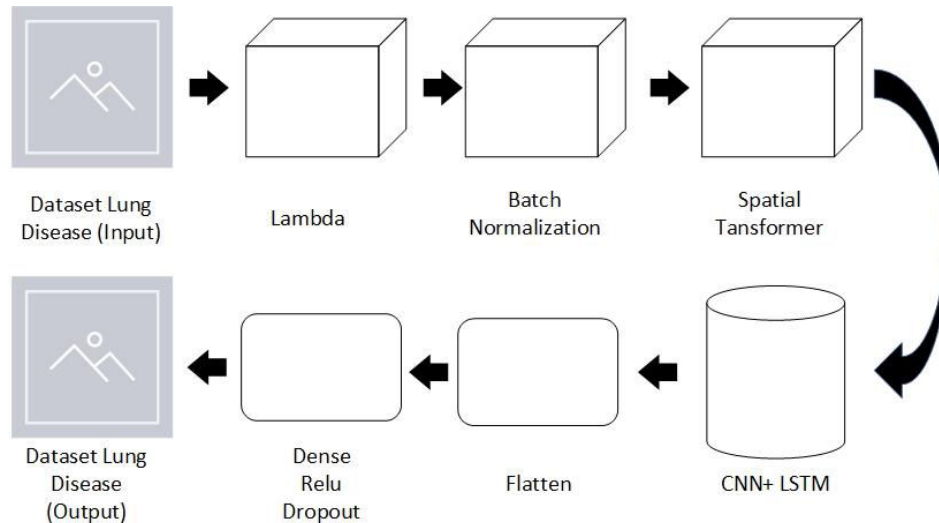


Figure 5. A Flowchart of Implementation CSLNet

Evaluation Metrics

To evaluate the proposed quantitative method's performance, evaluation metrics such as Accuracy (Acc), Precision (Pre), Recall (Rec), and F1-score are calculated statistically from the confusion matrix. Acc measures how often the classifier predicts correctly, while precision describes the positivity of cases predicted as positive. Precision is useful in cases where False Positives are more important than False Negatives. Meanwhile, recall describes how many actual positive cases can be predicted with the created model. Recall is useful when False Negatives are more important than False Positives. F1-score provides a combined idea of Precision and Recall metrics. The F1-score is maximum when Precision is equal to Recall, and the selected evaluation metrics are defined as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \times 100 \quad (7)$$

$$Recall = \frac{TP}{TP + FN} \quad (8)$$

$$Precision = \frac{TP}{TP + FP} \quad (9)$$

$$F1 - score = \frac{2 (Recall * Precision)}{2 (Recall + Precision)} \quad (10)$$

TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative. This study calculates Recall, Precision, and F1-score of the proposed method, and the formula for each metric is divided between equations (7)-(10). This metric shows the quality of the classification results and the weaknesses of the model for certain categories.

EXPERIMENTAL RESULT ANALYSIS AND DISCUSSION

Experimental Setup

In the experiment, the dataset was divided into 80%, 10%, and 10% for training, validation, and testing, respectively. The results were obtained using a 5-fold cross-validation technique and the proposed network consisted of 12 convolutional layers as shown in Table 3. Learning rate was 0.001 and the maximum number of epochs was 25, as determined experimentally.

CNN, CNN-LSTM, and CSLNet networks were implemented using Jupyter notebook, Python and the hardware library as well as TensorFlow on an Intel® Core™ i7-2.8GHz processor. Furthermore, tests were carried out using NVIDIA GTX 1060 graphics processing (GPU) with 6 and 16 GB RAM, respectively.

Model Training and Validation

The proposed model was trained and evaluated using augmented X-ray data. To demonstrate the model's convergence when the test dataset was limited, the proposed CSLNet was evaluated on k-fold cross-validation (k=5 folds). From all the collected samples in Table 1, the dataset distribution was shown in Table 2 and the model was evaluated on various ML classification metrics, including Accuracy, Precision, Recall, f1-score, and ROC. The experimental results comparing CNN, CNN+LSTM, and CSLNet models were shown in Figure 7 and Table 4.

Table 3. A Complete Summary of Proposed Model

Type	Kernel Size	Input Size
Input	-	224 x 224 x 3
Convolution2D	3 x 3	224 x 224 x 64
Convolution2D	3 x 3	224 x 224 x 64
Pool	2 x 2	112 x 112 x 64
Convolution2D	3 x 3	112 x 112 x 128
Convolution2D	3 x 3	112 x 112 x 128
Pool	2 x 2	56 x 56 x 128
Convolution2D	3 x 3	56 x 56 x 256
Convolution2D	3 x 3	56 x 56 x 256
Pool	2 x 2	28 x 28 x 256
Convolution2D	3 x 3	28 x 28 x 512
Convolution2D	3 x 3	28 x 28 x 512
Convolution2D	3 x 3	28 x 28 x 512
Pool	2 x 2	14 x 14 x 512
Convolution2D	3 x 3	14 x 14 x 1024
Convolution2D	3 x 3	14 x 14 x 1024
Convolution2D	3 x 3	14 x 14 x 1024
Pool	2 x 2	7 x 7 x 1024
LSTM	-	49 x 1024
Flatten	-	0
FC	-	51.381.248
Output	-	5125

Table 4. Model Comparison Results

Fold	CNN		CNN+LSTM		CSLNet	
	Average Accuracy	Average Loss	Average Accuracy	Average Loss	Average Accuracy	Average Loss
1	0.79	1.88	0.81	4.95	0.88	3.15
2	0.90	0.95	0.92	1.06	0.93	0.90
3	0.93	0.77	0.94	0.75	0.94	0.63
4	0.92	0.51	0.95	0.59	0.97	0.46
5	0.92	0.79	0.96	0.59	0.99	0.36

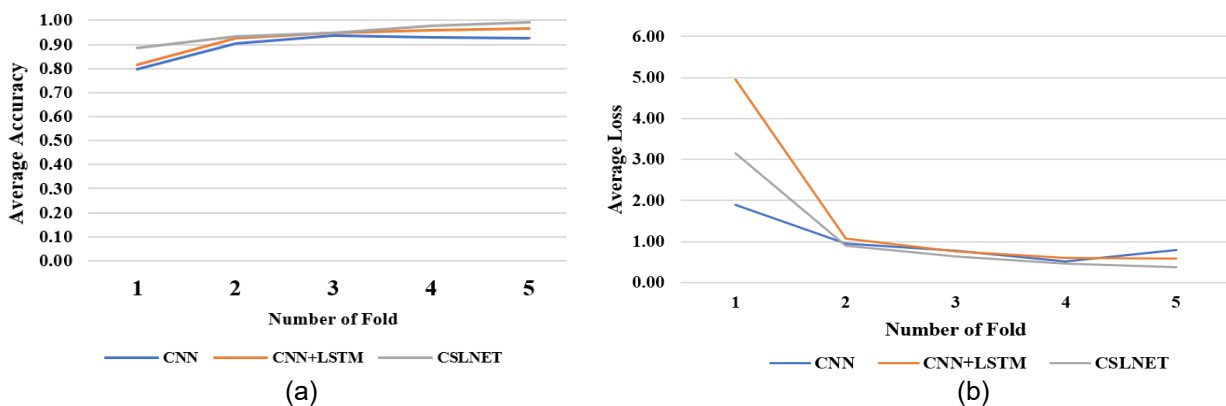


Figure 6. Learning Curve Visualization for (a) Test Accuracy and (b) Loss Accuracy with 5-Folds Cross Validation

Table 5. Performance Metrics (Accuracy, Precision, Recall, F1-score) for Each Test Device Class

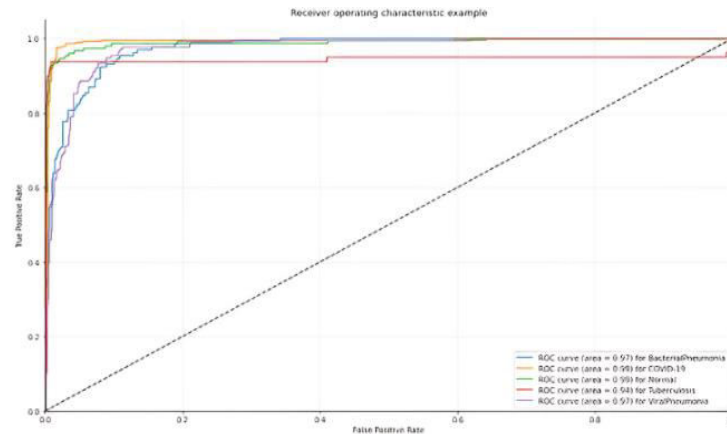
ID	Data	Accuracy	Precision	Recall	F1-score
1	Bacterial pneumonia	0.99	0.99	0.99	0.99
2	COVID-19	0.99	0.98	0.99	0.98
3	Normal	0.99	0.99	0.98	0.98
4	Tuberculosis	0.99	0.99	0.99	0.99
5	Viral pneumonia	0.99	0.99	0.99	0.99
6	Weight average	0.99	0.98	0.98	0.98

Figure 6 showed Accuracy and loss curves for the proposed CNN, CNN+LSTM, and CSLNet models. The 5 folds were run to train the models and Accuracy, as well as the loss for the training and validation datasets, were recorded simultaneously. To avoid overfitting on the training dataset, the models were trained only up to 25 epochs for each fold. The training and validation Accuracy of 0.98 and 0.99 were recorded after 5 folds, with loss values of 0.41 and 0.36, respectively. Learning curves showed that the best Accuracy and loss values were achieved by CSLNet model.

The performance of the proposed network was evaluated on independent test sets. Accuracy, Precision, Recall, and f1-score for each class were calculated and summarized in Table 5.

The results showed a significantly high Accuracy in the independent test data set for all classes, with a mean of 0.99, 0.98 Precision, 0.98 Recall, and 0.98 f1-score. Receiver Operating Characteristics (ROC) for bacterial pneumonia, COVID-19, normal, tuberculosis, and viral pneumonia were 0.97, 0.99, 0.99, 0.94, and 0.97, respectively as shown in Figure 7a and Figure 7b shows the confusion matrix.

Evaluation Metrics



(a) Receiver Operating Characteristics (ROC)

True label	1	128	0	0	1	1
	2	0	366	2	0	0
	3	2	0	147	1	0
	4	1	1	0	78	0
	5	0	1	0	0	175
		1	2	3	4	5
		Predicted label				

(b) Confusion Matrix

Figure 7. Classification Performance is Represented by (a) ROC, (b) Confusion Matrix, Class (1) Bacterial Pneumonia, (2) COVID-19, (3) Normal, (4) Tuberculosis, and (5) Viral Pneumonia

Lung Diseases Detection Results

After the created model successfully classified lung diseases based on their class, the model also detected diseases inside lung, as presented in Figure 8.

Discussion

This study designed and implemented a machine-learning model for diagnosing lung diseases from X-ray. The high performance, as shown in Table 6 suggested that the model can assist radiologists in their diagnoses. This provided a solution with reduced time and effort for diagnosing lung diseases and distinguishing healthy patients. The primary motivation of this study was to ensure easy implementation and use by radiologists when patients suffer from

lung disorders caused by diseases. To obtain a comprehensive dataset, a large number of chest X-ray were collected from the Kaggle repository, which included a total of 9039 different diseases. The number-based labeling was used due to its effectiveness in terms of cost and time.

Implementation of Lung Diseases Detection Tools

All coding was carried out using the Python programming language on Visual Studio Code with MySQL as the database. The login page as presented in Figure 9, was specifically designed for specialist doctors to access the start of the system.

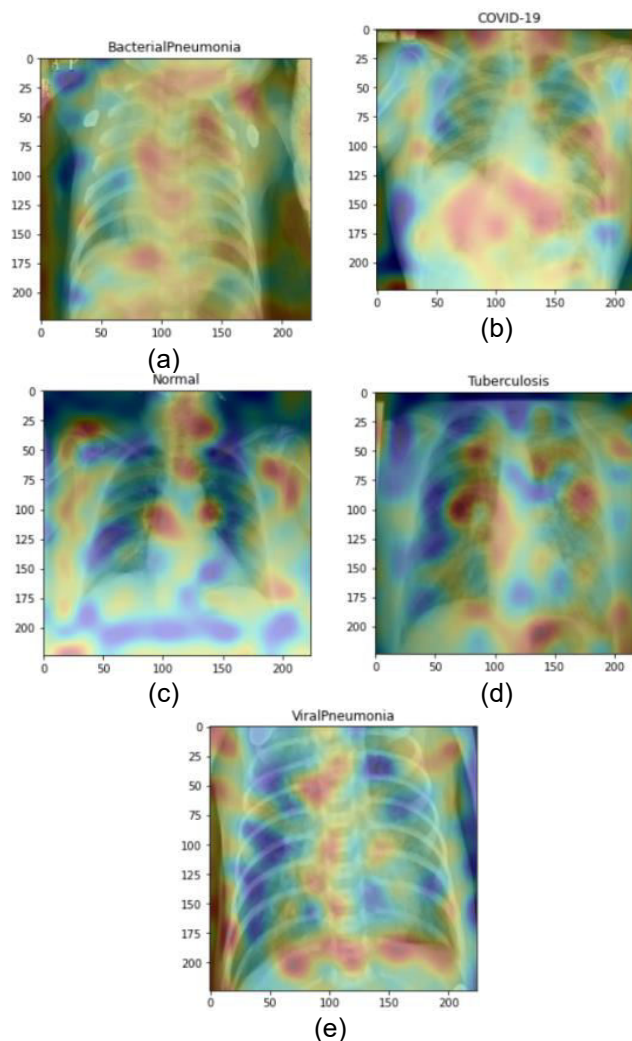


Figure 8. Result of Detection Lung Disease Detection (a) Bacterial Pneumonia, (b) COVID-19, (c) Normal, (d) Tuberculosis, and (e) Viral Pneumonia

Table 6. Comparison of Architectural Models Used in Related Studies and This Study

Study	Feature Extraction Model	Sequence Model	Accuracy
[3]	VGG	-	73%
[11]	Conv2d	-	98%
[22]	Conv2d	-	93.3%
[24]	ResNet 50	-	96.1%
[25]	Conv2d	-	93.3%
This work	STN and CNN	LSTM	99%

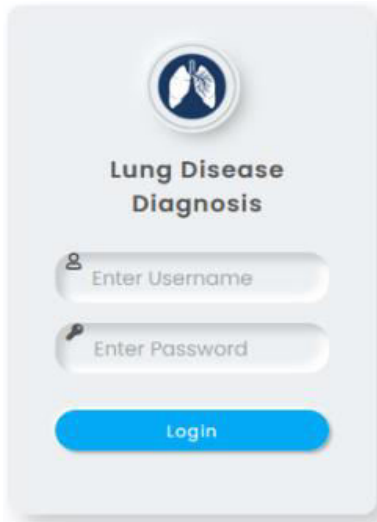


Figure 9. Login Page

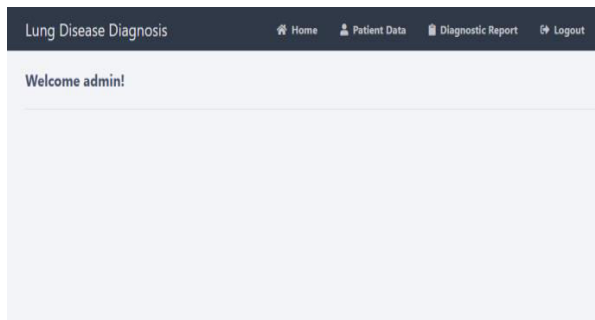


Figure 10. Home Page

The radiologist must enter a username and password before accessing lung diagnostic system. When the radiologist entered the username and password incorrectly, the system will reject the login request. However, when correctly entered, the system will approve the radiologist's access to the system. Figure 10 showed the home page when the doctor successfully logged in.

Radiologists must also enter patient data first before diagnosing diseases of the patient. The patient data that should be entered by the radiologist included the name, age, gender, contact, and address of the patient. Figure 11 showed the results of patient data input entered by the radiologist.

When a radiologist entered the results of a chest X-ray images, the radiologist can immediately predict diseases of the patient by pressing the prediction button. Furthermore, the radiologist can save and print the entered diagnosis report to the database by pressing the submit button. Figure 12 showed the results of diseases diagnosis input performed by the radiologist.

Diagnostic reports carried out by radiologists were stored in a database and became output reports that were given to patients. Figure 13 showed the output report on the results of a lung diseases diagnosis.

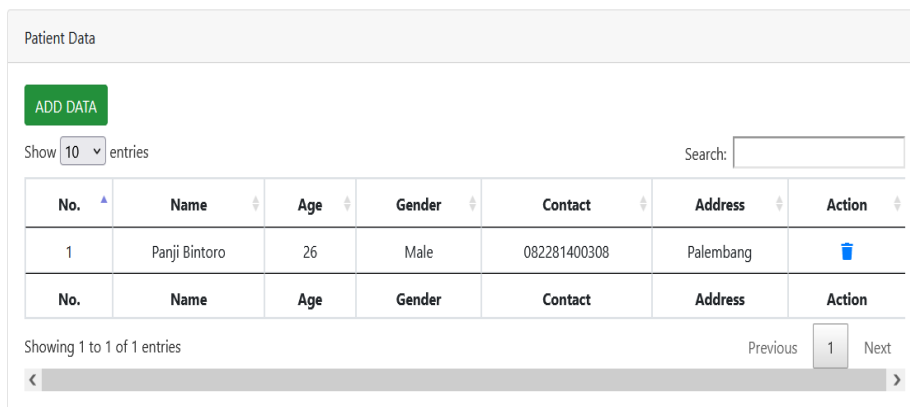




Figure 11. Patient's Data Page

Diagnostic Report

[ADD DATA](#)

Show entries Search:

No. [▲]	Name [⚡]	Age [⚡]	Gender [⚡]	Contact [⚡]	Address [⚡]	diagnostic results [⚡]	Action [⚡]
1	Panji Bintoro	26	Male	082281400308	Palembang	COVID-19	 

Showing 1 to 1 of 1 entries Previous Next

< >

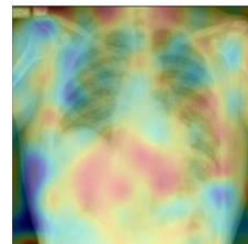
Figure 12. Diagnostics Report Page

Lung Disease Diagnosis Result

Name : Panji Bintoro Contact : 082281400308
 Age : 26 Address : Palembang
 Gender : Male



Diagnostic Result: COVID-19



Lung Disease Detection Results

Figure 13. Output Report on The Results of a Lung Diseases Diagnosis

CONCLUSION

This study proposed a new deep learning called CSLNet to detect lung diseases from X-ray images. The new model was applied to a dataset of chest X-ray images collected from the Kaggle repository. The CNN and CNN+LSTM showed the best average validation Accuracy of 0.925 and 0.965, respectively. Meanwhile, the proposed model, namely CSLNet, showed the best average validation Accuracy of 0.992. In terms of evaluation metrics, CSLNet showed promising results with an average of 0.99 Accuracy, 0.98 Precision, 0.98 Recall, and an f1-score of 0.98. ROC for bacterial pneumonia,

COVID-19, normal, tuberculosis, and viral pneumonia were 0.97, 0.99, 0.99, 0.94, and 0.97, respectively.

CONFLICT OF INTEREST

The authors declare that they have no knowledge of any financial conflicts of interest or personal relationships that could have influenced the work reported in this paper.

REFERENCES

[1] S. Safiri et al., "Burden of chronic obstructive pulmonary disease and its attributable risk factors in 204 countries and territories, 1990-2019: Results from the

- Global Burden of Disease Study 2019,” *BMJ*, 2022, doi: 10.1136/bmj-2021-069679.
- [2] F. Hussein et al., “Hybrid CLAHE-CNN Deep Neural Networks for Classifying Lung Diseases from X-ray Acquisitions,” *Electronics*, vol. 11, no. 19, p. 3075, 2022, doi: 10.3390/electronics11193075.
- [3] S. Bharati, P. Podder, and M. R. H. Mondal, “Hybrid deep learning for detecting lung diseases from X-ray images,” *Informatics Med. Unlocked*, vol. 20, p. 100391, 2020, doi: 10.1016/j.imu.2020.100391.
- [4] M. Oloko-Oba and S. Viriri, *Diagnosing tuberculosis using deep convolutional neural network*, vol. 12119 LNCS. Springer International Publishing, 2020. doi: 10.1007/978-3-030-51935-3_16.
- [5] World Health Organization, “Laboratory testing for coronavirus disease 2019 (COVID-19) in suspected human cases,” vol. 2019, no. March, 2020.
- [6] S. M. Fati, E. M. Senan, and N. ElHakim, “Deep and Hybrid Learning Technique for Early Detection of Tuberculosis Based on X-ray Images Using Feature Fusion,” *Appl. Sci.*, vol. 12, no. 14, p. 7092, 2022, doi: 10.3390/app12147092.
- [7] M. Z. Alom, M. M. S. Rahman, M. S. Nasrin, T. M. Taha, and V. K. Asari, “COVID_MTNNet: COVID-19 Detection with Multi-Task Deep Learning Approaches,” 2020, [Online]. Available: <http://arxiv.org/abs/2004.03747>
- [8] M. F. Aslan, “A robust semantic lung segmentation study for CNN-based COVID-19 diagnosis,” *Chemom. Intell. Lab. Syst.*, vol. 231, no. October, p. 104695, 2022, doi: 10.1016/j.chemolab.2022.104695.
- [9] M. Heidari, S. Mirniaharikandehi, A. Z. Khuzani, G. Danala, Y. Qiu, and B. Zheng, “Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms,” *Int. J. Med. Inform.*, vol. 144, no. September, p. 104284, 2020, doi: 10.1016/j.ijmedinf.2020.104284.
- [10] R. Karthik, R. Menaka, and M. Hariharan, “Learning distinctive filters for COVID-19 detection from chest X-ray using shuffled residual CNN,” *Appl. Soft Comput.*, vol. 99, p. 106744, 2021, doi: 10.1016/j.asoc.2020.106744.
- [11] S. Thakur and A. Kumar, “X-ray and CT-scan-based automated detection and classification of covid-19 using convolutional neural networks (CNN),” *Biomed. Signal Process. Control*, vol. 69, no. March, p. 102920, 2021, doi: 10.1016/j.bspc.2021.102920.
- [12] A. Narula and N. K. Vaegae, “Development of CNN-LSTM combinational architecture for COVID-19 detection,” *J. Ambient Intell. Humaniz. Comput.*, no. 0123456789, 2022, doi: 10.1007/s12652-022-04508-2.
- [13] A. Irfan, A. L. Adivishnu, A. Sze-To, T. Dehkharghanian, S. Rahnamayan, and H. R. Tizhoosh, “Classifying Pneumonia among Chest X-Rays Using Transfer Learning,” *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 2020-July, pp. 2186–2189, 2020, doi: 10.1109/EMBC44109.2020.9175594.
- [14] F. Ucar and D. Korkmaz, “COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease 2019 (COVID-19) from X-ray images,” *Med. Hypotheses*, vol. 140, no. April, p. 109761, 2020, doi: 10.1016/j.mehy.2020.109761.
- [15] D. S. Kermany et al., “Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning,” *Cell*, vol. 172, no. 5, pp. 1122–1131.e9, 2018, doi: 10.1016/j.cell.2018.02.010.
- [16] D. Varshni, K. Thakral, L. Agarwal, R. Nijhawan, and A. Mittal, “Pneumonia Detection Using CNN based Feature Extraction,” *Proc. 2019 3rd IEEE Int. Conf. Electr. Comput. Commun. Technol. ICECCT 2019*, 2019, doi: 10.1109/ICECCT.2019.8869364.
- [17] Ü. Budak, Z. Cömert, M. Çıbuk, and A. Şengür, “DCCMED-Net: Densely connected and concatenated multi Encoder-Decoder CNNs for retinal vessel extraction from fundus images,” *Med. Hypotheses*, vol. 134, no. August 2019, 2020, doi: 10.1016/j.mehy.2019.109426.
- [18] F. Özyurt, E. Sert, and D. Avcı, “An expert system for brain tumor detection: Fuzzy C-means with super resolution and convolutional neural network with extreme learning machine,” *Med. Hypotheses*, vol. 134, no. September 2019, 2020, doi: 10.1016/j.mehy.2019.109433.
- [19] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size,” pp. 1–13, 2016, [Online]. Available: <http://arxiv.org/abs/1602.07360>
- [20] G. Fan, F. Chen, D. Chen, and Y. Dong, “Recognizing multiple types of rocks quickly and accurately based on lightweight CNNs model,” *IEEE Access*, vol. 8, pp. 55269–55278, 2020, doi: 10.1109/ACCESS.2020.2982017.
- [21] R. T. N. V. S. Chappa and M. El-Sharkawy, “Squeeze-and-Excitation SqueezeNext: An

- Efficient DNN for Hardware Deployment,” 2020 10th Annu. Comput. Commun. Work. Conf. CCWC 2020, pp. 691–697, 2020, doi: 10.1109/CCWC47524.2020.9031119.
- [22] L. Wang, Z. Q. Lin, and A. Wong, “COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images,” *Sci. Rep.*, vol. 10, no. 1, pp. 1–12, 2020, doi: 10.1038/s41598-020-76550-z.
- [23] I. D. Apostolopoulos and T. A. Mpesiana, “Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks,” *Phys. Eng. Sci. Med.*, vol. 43, no. 2, pp. 635–640, 2020, doi: 10.1007/s13246-020-00865-4.
- [24] A. Narin, C. Kaya, and Z. Pamuk, “Department of Biomedical Engineering, Zonguldak Bulent Ecevit University, 67100, Zonguldak, Turkey.,” *arXiv Prepr. arXiv2003.10849.*, 2020, [Online]. Available: <https://arxiv.org/abs/2003.10849>
- [25] Y. Pathak, P. K. Shukla, A. Tiwari, S. Stalin, and S. Singh, “Deep Transfer Learning Based Classification Model for COVID-19 Disease,” *Irbm*, vol. 43, no. 2, pp. 87–92, 2022, doi: 10.1016/j.irbm.2020.05.003.
- [26] Khandar, “Tuberculosis (TB) Chest X-ray Database.” [Online]. Available: <https://www.kaggle.com/datasets/tawsifurrahman/tub>
- [27] Jtiptj, “Chest X-Ray (Pneumonia, Covid-19, Tuberculosis).” [Online]. Available: <https://www.kaggle.com/datasets/jtiptj/chest-xray->
- [28] J. Simonsen and O. S. Jensen, “Spatial Transformer Networks,” *ACM Int. Conf. Proceeding Ser.*, vol. 2, pp. 45–48, 2016, doi: 10.1145/2948076.2948084.
- [29] N. M. Elshennawy and D. M. Ibrahim, “Deep-Pneumonia Framework Using Deep Learning Models Based on Chest X-Ray Images,” *Diagnostics*, vol. 10, no. 9, pp. 1–16, 2020, doi: 10.3390/diagnostics10090649.
- [30] G. Chen, “A Gentle Tutorial of Recurrent Neural Network with Error Backpropagation,” pp. 1–10, 2016, [Online]. Available: <http://arxiv.org/abs/1610.02583>
- [31] T. Szandała, “Review and comparison of commonly used activation functions for deep neural networks,” *Stud. Comput. Intell.*, vol. 903, pp. 203–224, 2021, doi: 10.1007/978-981-15-5495-7_11.