Pneumonia Detection from X-Ray Images Using Deep Learning

1st Albany Siswanto
BINUS University
Bandung, Indonesia
albany.siswanto@binus.ac.id

2nd Muhammad Rafi Isnaen
BINUS University
Bandung, Indonesia
muhammad.isnaen@binus.ac.id

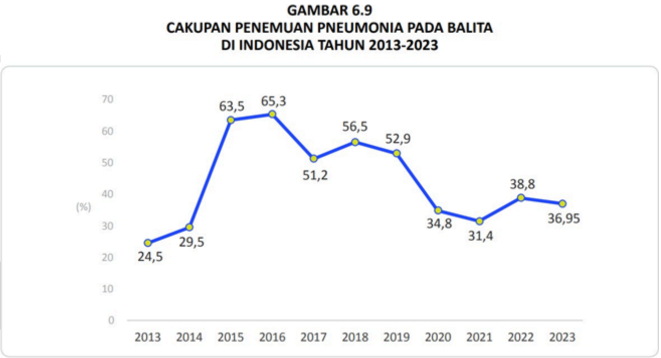
3rd Naufal Ghifari Hidayat
BINUS University
Bandung, Indonesia
naufal.hidayat001@binus.ac.id

***Abstract*—Pneumonia remains a significant health challenge, particularly in children under five and the elderly, due to its high morbidity and mortality rates. Traditional diagnostic methods, relying on chest X-rays and clinical evaluation, are time-consuming and prone to errors, especially in resource-limited settings. This study proposes an artificial intelligence (AI)-based diagnostic system utilizing Convolutional Neural Networks (CNNs) to automate pneumonia detection from chest X-ray images. The model trained with over 5000 labeled images, achieves high accuracy by identifying intricate patterns indicative of pneumonia, including opacities and consolidations. By integrating this system into clinical workflows, healthcare providers can enhance diagnostic efficiency, reduce human error, and improve patient outcomes. The findings demonstrate the potential of AI to revolutionize pneumonia detection, addressing critical gaps in global healthcare delivery.**

***Index Terms*—pneumonia detection, AI, CNNs, chest x-ray analysis, deep learning.**

I. INTRODUCTION

Pneumonia remains a leading cause of mortality, particularly among children under five years old and the elderly. According to the World Health Organization (WHO), this respiratory infection poses significant challenges in areas with poor air quality and limited access to healthcare services. Delays in diagnosis and treatment often lead to severe complications, emphasizing the need for rapid and accurate diagnostic tools.



Sumber: Ditjen P2P, Kemenkes RI, 2024

Fig. 1. Pneumonia coverage in children under five in a decade

Traditional diagnostic methods rely on chest X-rays combined with physical examinations and anamnesis, requiring experienced medical professionals to interpret the results accurately. However, the high volume of cases and limited resources in healthcare systems often result in diagnostic delays and human errors. Recent advancements in artificial intelligence (AI), particularly deep learning algorithms like Convolutional Neural Networks (CNN), offer promising solutions to enhance the efficiency and precision of pneumonia detection.

This study aims to develop an AI-based diagnostic system capable of analyzing chest X-ray images to identify pneumonia with high accuracy. Leveraging a dataset of over 5,000 labeled images, the proposed model employs CNN to detect critical patterns and anomalies indicative of pneumonia. The integration of such AI systems into clinical workflows has the potential to improve diagnostic accuracy, reduce the workload on healthcare professionals, and ultimately enhance patient outcomes.

By bridging the gap between traditional diagnostic practices and modern AI capabilities, this research seeks to contribute to the broader application of AI in medical imaging, particularly in resource-constrained settings. The results of this study are expected to highlight the practical benefits of AI in addressing global health challenges.

II. LITERATURE REVIEW

2.1. Characteristics of Pneumonia

Pneumonia is characterized by inflammation and infection in the lung tissue, particularly in the alveoli, leading to the accumulation of fluid or pus. This condition is commonly identified through X-ray imaging, which reveals white opacities indicating alveolar consolidation. The causes of pneumonia can vary, including bacterial infections that often present with lobar infiltrates and viral infections that display diffuse patterns like ground-glass opacities.

Clinically, pneumonia manifests with symptoms such as cough, fever, chest pain, and difficulty breathing. In severe cases, pleural effusion and respiratory failure may occur, highlighting the need for timely and accurate diagnosis. The combination of physical examination, imaging, and laboratory tests forms the cornerstone of traditional diagnostic approaches, although these methods are time-intensive and dependent on the availability of skilled medical professionals.

2.2. Declining Health Outcomes in Pneumonia Patients

Deterioration of health in pneumonia patients occurs when detection and treatment are delayed. If left untreated, pneumonia can develop into more severe conditions, including respiratory failure or sepsis. The decline in health is also influenced by the availability of health facilities and the ability of medical personnel to detect the disease early. AI technology is expected to help speed up diagnosis so that the quality of treatment improves and mortality decreases.

2.3. Application of Artificial Intelligence in Healthcare

The use of artificial intelligence in healthcare has been increasing, especially in the processing of medical images such as X-rays and CT scans. AI is able to analyze patterns in images quickly and accurately, and detect anomalies that may be missed by medical personnel. In the context of pneumonia, AI can be used to speed up the disease identification process from X-ray images and provide initial recommendations, which are then confirmed by doctors. Such a system is already being implemented in some hospitals and is expected to become the standard in image-based diagnosis.

2.4. Using CNN Deep Learning to analize images

Convolutional Neural Networks (CNNs) are deep learning algorithms designed for grid-like data, such as images. They use convolutional layers to extract visual features like edges, textures, and shapes, while pooling layers reduce data dimensions to improve computational efficiency. CNNs automatically learn hierarchical patterns from data, making them highly effective for image-based tasks without requiring complex manual preprocessing.

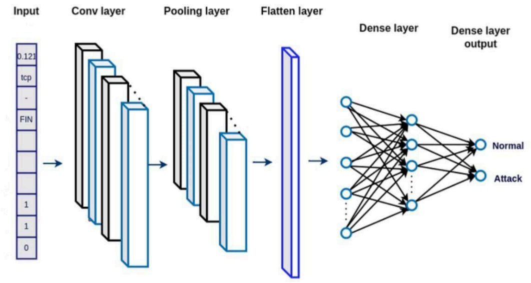


Fig. 2. Layers in CNN Model

Convolutional Layer: Captures important visual patterns through convolution operations, such as detecting edges, textures, or other features from the input data.

Pooling Layer: Reduces the dimensionality of the data using methods such as max pooling or average pooling, thereby reducing the number of parameters, speeding up computation, and improving robustness to small shifts in the data.

Flatten Layer: Converts the output data from a matrix or tensor format to a 1-dimensional vector so that it can be processed by the next layer.

Dense Layer (Fully Connected Layer): Combines all the extracted features into a final representation to generate a prediction, such as a classification (e.g. “Normal” or “Attack”).

The main advantage of CNNs in image analysis lies in their ability to automatically learn features from data, making CNN highly effective for handling complex visual data. Unlike traditional methods that require manual feature design, CNNs can identify detailed patterns that are difficult for humans to recognize. This makes them a superior choice in applications such as image classification, object detection, and image segmentation. In addition, CNNs are also highly flexible as they can be adapted to different types of data and tasks, either through architecture modification or transfer learning. With the ability to utilize large training data, CNNs are able to deliver accurate and consistent results, making them an ideal technology for various image-based applications.

III. METHODOLOGY

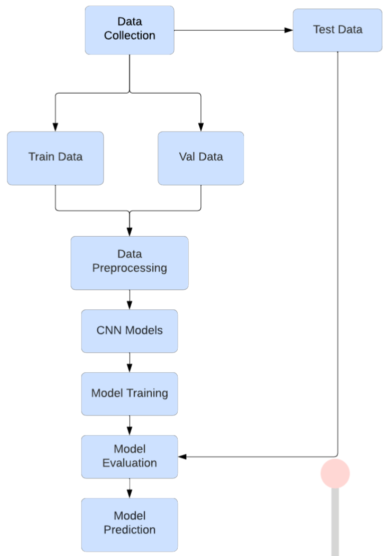


Fig. 3. Workflow Flowchart

3.1. Data Collection

The dataset used in this study was obtained from Kaggle, specifically the "Chest X-ray Pneumonia" dataset, which contains 5,856 labeled chest X-ray images. The data is categorized into two classes: pneumonia and normal. The dataset was split into training, validation, and testing sets as follows:

Training set: 5,216 images (1,341 normal, 3,875 pneumonia)

Validation set: 16 images (8 normal, 8 pneumonia)

Testing set: 624 images (234 normal, 390 pneumonia)

The data was carefully validated to ensure the images were of sufficient quality, correctly labeled, and free of corruption or noise.

3.2. Data Preprocessing

The image is first normalized by scaling its pixel value parameter to a range of 0 to 1. Other Augmentation parameters include random rotation of up to 10 degrees, horizontal and vertical shift of up to 10% each of the image dimensions, image tilt of up to 10%, as well as random zoom of up to 15%. In addition, the image can also be flipped horizontally to reflect a different orientation. Blank areas that appear due to the transformation are filled using nearby pixels.

The data processing process also includes preparing a data generator for the hard framework to efficiently load the dataset during training and validation. In this case, image normalization by resizing all images to 224x224 pixels, according to the model input requirements. Next, the batch parameter is set to 32, specifying the number of images to be processed at once in a batch, which helps in memory management during training. With the class mode parameter, the data is classified into two classes (binary classes), namely pneumonia and normal. This pre-processing technique helps the model to learn image variations, thus improving the generalization ability of the model to new data.

3.3. Model Development

A CNN-based model was designed using Python and the TensorFlow framework, employing the following architecture:

Convolutional Layers: Three layers with increasing filter sizes (32, 64, and 128), each using a kernel size of 3x3 and ReLU activation.

Pooling Layers: Max-pooling layers with a 2x2 filter to downsample the spatial dimensions of feature maps.

Flattening Layer: Converts the 2D feature maps into a 1D vector.

Dense Layers:

1. A fully connected layer with 128 neurons and ReLU activation.
2. An output layer with one neuron and sigmoid activation to classify images into pneumonia or normal categories.

The model was optimized using the Adam optimizer with a learning rate of 0.001. The loss function used was binary cross-entropy. Training was conducted over 5 epochs with a batch size of 32.

3.4. Model Evaluation

The model's performance was assessed using the following metrics:

Accuracy: Proportion of correct predictions over the total predictions.

Precision: Fraction of true positive predictions among all positive predictions.

Recall: Fraction of true positives correctly identified out of all actual positives.

F1-Score: Harmonic mean of precision and recall, used to balance the trade-off between the two.

3.5. Image Prediction

The trained model was used to predict images by calculating the probability of pneumonia or normal classes. The prediction threshold was set at 0.5, with the confidence score calculated as follows:

For probabilities > 0.5: $Confidence = Prediction \times 100$

For probabilities ≤ 0.5: $Confidence = (1 - Prediction) \times 100$

The final system was deployed on a web-based interface to facilitate user interaction, allowing users to upload images and view prediction results with confidence score.

IV. CNN MODEL PERFORMANCE EVALUATION

Model performance evaluation needs to be done to determine the potential success of machine learning models to make predictions. Using techniques such as test accuracy, confusion matrix, and others.

4.1. Test Accuracy

Using a test dataset of 624 data. The test dataset that has been provided directly separately, is not used in the model training process. Its function is to provide an objective picture of how well the trained model can generalize to data that has never been seen before.

Once the model has been trained using the training and validation data, the testset dataset is fed to the model. Accuracy is calculated as the proportion of correct predictions to total predictions, with the formula :

$$Accuracy = \frac{Number\ of\ Correct\ Accuracy}{Total\ Prediction}$$

This evaluation method can be applied to the program by using model.evaluate(), with the model that has been created.

4.2. Confusion Matrix

Confusion matrix is a simple evaluation method to see the performance of a classification model. This evaluation is in the form of a table represented by a matrix. The confusion matrix table shows what the model guessed compared to the actual truth (original label). By using the confusion matrix, the model can be seen how often it is right or wrong in making predictions.

For a binary classification problem (2 classes: Positive and Negative), the table can be described like this:

TABLE I
Confusion Matrix Evaluation Table

Prediction \ Actual	Positive (Original)	Negative (Original)
Positive (Prediction)	True Positive (TP)	False Positive (FP)
Negative (Prediction)	False Negative (FN)	True Negative (TN)

- True Positive (TP): The model predicts “Positive” and is correct.
- True Negative (TN): The model predicts “Negative” and is correct.
- False Positive (FP): The model predicts “Positive”, but is wrong (aka “false alarm”).
- False Negative (FN): The model predicts “Negative”, but is wrong (aka “missed”).

4.3. Precision

Precision is how many of the model's “Positive” predictions are actually positive. That is, precision helps answer: “Of all the predicted positives, how accurate is the prediction of the people who are actually positive?”

$$Precision = \frac{True\ Positive(TP)}{True\ Positive\ (TP)+False\ Positive\ (FP)}$$

By way of example, if the model predicts 10 sick people, but only 8 are actually sick, then precision = 8/10 = 0.8 (80%).

4.4. Recall

Recall is how many positive cases the model managed to find. That is, recall answers: “Of all the true positives, how many were found by the model?”

$$Recall = \frac{True\ Positive(TP)}{True\ Positive\ (TP)+False\ Negative\ (FN)}$$

With a case example, if there are 20 sick people, and the model only detects 15, then recall = 15/20 = 0.75 (75%).

4.5. F1-Score

F1-Score is the average between precision and recall. This means that F1-Score is used to balance precision and recall, especially if both are important.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision+Recall}$$

4.6. Support

Support is a number that indicates the amount of original data in each class (Positive or Negative). That is, support simply calculates how many actual samples there are for each class.

Since support evaluates the model using the test dataset, there are a total of 624 samples, with negative support samples = 234 and positive support = 390.

4.7. Average

In the model evaluation, macro average and weighted average are used to provide a comprehensive overview of the model's performance when the dataset has an unbalanced class distribution.

4.7.1. Macro Average

Macro Average calculates the metric for each class separately, and then takes the average regardless of the class size.

$$Macro\ Average = \frac{metric\ class\ 1 + metric\ class\ 2 + ... + metric\ class\ n}{number\ of\ classes}$$

With an example, if the Precision of class A = 0.90 and the Precision of class B = 0.90, then Macro Average = (0.90 + 0.90) / 2 = 0.90 (90%).

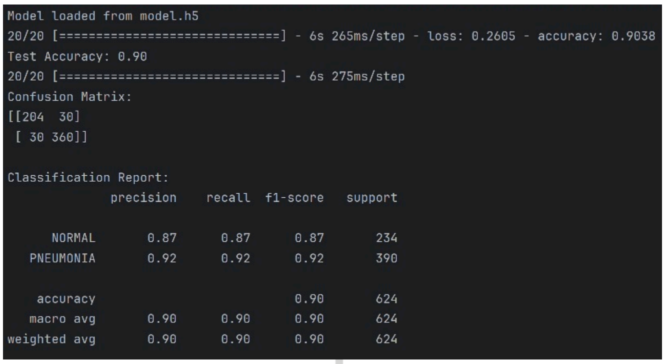
4.7.2. Weighted Average

Weighted Average calculates the metric for each class separately, but each metric is weighted according to the size of the class. Weighted Average takes into account the distribution of the data.

$$Weighted\ Average = \frac{(Metrik\ kelas\ 1 \times Jumlah\ data\ kelas\ 1)+(Metrik\ kelas\ 2 \times Jumlah\ data\ kelas\ 2)+...}{Total\ Jumlah\ Data}$$

By way of example, if Class A Precision = 0.90 (50 data), Class B Precision = 0.50 (30 data), then Weighted Average = ((0.90 x 50) + (0.50 x 30)) / 80 = 0.75 (75%).

V. RESULT AND DISCUSSION



```
Model loaded from model.h5
20/20 [==============================] - 6s 265ms/step - loss: 0.2605 - accuracy: 0.9038
Test Accuracy: 0.90
20/20 [==============================] - 6s 275ms/step
Confusion Matrix:
[[204  30]
 [ 30 360]]

Classification Report:
              precision    recall  f1-score   support

      NORMAL       0.87      0.87      0.87       234
   PNEUMONIA       0.92      0.92      0.92       390

    accuracy                           0.90       624
   macro avg       0.90      0.90      0.90       624
weighted avg       0.90      0.90      0.90       624
```

Fig. 4. Trained Model Evaluation Results

from the image of Fig. 4, recorded results are :

- **Test Accuracy :** 0.90 (90%)
- **Confusion Matrix :**

TABLE II
CONFUSION MATRIX OF TRAINED MODEL

Prediction \ Actual	Positive (Original)	Negative (Original)
Positive (Prediction)	204	30
Negative (Prediction)	30	360

- **Classification Report :**

TABLE III
CLASSIFICATION REPORT OF TRAINED MODEL

Metrics	Precision	Recall	F1-Score	Support
Normal	0.87	0.87	0.87	234
Pneumonia	0.92	0.92	0.92	390
Accuracy	-	-	0.90	624
Macro Avg.	0.90	0.90	0.90	624
Weighted Avg.	0.90	0.90	0.90	624

From results provided, the pneumonia detection AI model showed excellent performance with an overall accuracy of 90%. In the Confusion Matrix, it can be seen that out of 390 patients with pneumonia (true positives), the model successfully detected 360 cases correctly, resulting in a recall value for pneumonia of 0.92 (92%). This indicates that of all patients who actually had pneumonia, 92% were successfully identified by the model.

5.1. Focused Metrics

With the pneumonia detection AI model created as a doctor's second assistant, the focus on Recall accuracy is very important while taking into account the medical context.

If the model fails to detect a patient with pneumonia (False Negative), the patient will not receive the necessary medical attention, resulting in a fatal outcome. Therefore, the model should minimize FN, and this is reflected in recall. The recall value answers: “ Of all the patients who were really sick, how many did the model successfully detect? ”

In a medical context, it is better to generate a few false positives (FP) (e.g., detecting pneumonia in a healthy patient) rather than to miss a truly sick patient.

VI. CONCLUSION

This study demonstrates the efficacy of AI-powered diagnostic systems in pneumonia detection. Utilizing Convolutional Neural Networks (CNNs), the proposed model achieved a high accuracy of 90% in detecting pneumonia from normal chest X-rays. This performance underscores the potential of integrating AI into clinical workflows to enhance diagnostic precision, reduce human error, and expedite decision-making processes. Future developments could explore expanding the dataset, incorporating multimodal diagnostics, and deploying the system in real-world healthcare environments to evaluate its scalability and robustness.

REFERENCES

[1] Pesheva, E. (2024). Does AI Help or Hurt Human Radiologists’ Performance? It Depends on the Doctor. Harvard Medical School. https://hms.harvard.edu/news/does-ai-help-or-hurt-human-radiologists-per formance-depends-doctor.

[2] Armitage, H. (2018). Artificial intelligence rivals radiologists in screening X-rays for certain diseases. Standford Medicine. https://med.stanford.edu/news/all-news/2018/11/ai-outperformed-radiologi sts-in-screening-x-rays-for-certain-diseases.html.

[3] Ait Nasser, A., & Akhloufi, M. A. (2023). Deep learning methods for chest disease detection using radiography images. SN Computer Science, 4(388). https://doi.org/10.1007/s42979-023-01818-w.

[4] Fahad, N., Ahmed, R., Jahan, F., Abdullah-Al-Jubair, M., & Morol, M. K. (2024). MIC: Medical image classification using chest X-ray (COVID-19 and pneumonia) dataset with the

help of CNN and customized CNN. Faculty of Information Science and Technology, Multimedia University, Melaka, Malaysia. https://doi.org/10.48550/arXiv.2411.01163.

[5] Meliboev, A., Alikhanov, J., & Kim, W. (2023). Performance Evaluation of Deep Learning Based Network Intrusion Detection System across Multiple Balanced and Imbalanced Datasets. MDPI. 11(4), 515. https://doi.org/10.3390/electronics11040515.