ISSN: 2581-1274

Deep Learning Based Classification of TB and Normal Chest X-rays Using a Custom CNN with Minimal Epoch Training

Zharifah Muthiah [a,1,*], Oktalia Triananda Lovita [a,2], Mohd Iqbal Muttaqin [a,3]

[a] *Universitas Bina Bangsa Getsempena, Jl. Tanggul Kreung Lamnyong, Banda Aceh and 23112, Indonesia*
[1] *zharifah@bbg.ac.id*;* [2] *oktalia@bbg.ac.id;* [3] *iqbalmuttaqin@bbg.ac.id*
** corresponding author*

ARTICLE INFO

Article history:
Published
July 14, 2025

Keywords:
Deep learning
Medical imaging
Custom Layer
CNN
Tuberculosis

ABSTRACT

Tuberculosis (TB) is a major global health concern and remains one of the deadliest infectious diseases, particularly in developing countries. Early and accurate diagnosis is crucial to initiate timely treatment, prevent complications, and reduce transmission rates. Conventional diagnostic methods, such as sputum tests and laboratory cultures, are often time-consuming and require specialized resources. Therefore, there is a growing need for automated, efficient, and accurate computer-aided diagnosis (CAD) systems. This study proposes a lightweight Convolutional Neural Network (CNN) architecture to classify chest X-ray images into TB and normal categories. The model is trained using the publicly available Shenzhen chest X-ray dataset, with three training durations: 10, 25, and 50 epochs. Although the model trained for 25 epochs achieved a slightly higher accuracy (86.36%) compared to the 10 epochs model (85.61%), the latter is considered more optimal due to its better balance between performance and efficiency. Specifically, the 10 epochs model produced high precision (92.86%) and a competitive F1-score (84.27%) while requiring significantly less training time and computational resources. Moreover, it maintained stable validation performance without signs of overfitting. In contrast, models trained for longer durations showed diminishing returns or performance degradation, particularly at 50 epochs. These results indicate that a shorter training cycle, when coupled with appropriate architectural design and regularization, can yield a robust and efficient classification model. This approach is particularly beneficial for deployment in resource-constrained environments, where rapid and reliable TB screening using chest X-ray images is critically needed.

I. Introduction

Tuberculosis (TB) is an infectious disease caused by *Mycobacterium tuberculosis*, and it remains one of the leading causes of death due to infectious diseases globally, particularly in developing countries such as Indonesia. According to the World Health Organization (WHO), more than 10 million individuals are infected with TB each year, with over 1 million fatalities recorded [1]. One of the main screening methods for TB is through chest X-ray (CXR) imaging due to its rapid, low-cost, and widely accessible nature, even at the primary healthcare level. However, manual interpretation of CXR images requires high expertise and experience from radiologists. The limited availability of experts and the subjective nature of interpretation often hinder early and accurate diagnosis of TB [2].

Deep learning approaches have been widely applied to address various tasks. For instance, the CNN-based model presented in[3] demonstrates remarkable performance in the classification and detection of different fish species on a large scale. Artificial intelligence technologies have also become integral to modern healthcare services, including disease detection through medical imaging [4]. The application of Computer-Aided Diagnosis systems[5], [6], has been shown to enhance the decision making capabilities of doctors and radiologists, thereby supporting more accurate healthcare delivery to patients.

CNN is among the most commonly used deep learning methods in medical image analysis due to its ability to extract spatial features from images without manual feature engineering, as required in traditional machine learning algorithms [7], [8]. With the availability of large-scale annotated datasets [9], along with hardware advancements such as GPUs that enable high-speed computation, studies on pulmonary disease detection through DL-based CXR analysis have achieved improved accuracy levels [10].

This study focuses on chest X-ray imaging as the primary object for detecting tuberculosis. These images are processed as RGB inputs with a resolution of 224×224×3 pixels. The subject of this study involves the development of a custom Convolutional Neural Network (CNN) designed to classify these images into two distinct categories: "TB" or "Normal". The research is conducted through computational modeling and analysis, without a specific physical location, as it primarily involves software development and data processing.

The analysis reveals that the custom CNN offers a viable solution for automated TB detection using chest X-ray images [11]. This approach can facilitate rapid and consistent preliminary diagnosis, potentially reducing the workload of medical practitioners and accelerating diagnostic processes in resource-limited settings. The inclusion of multiple dropout layers within the network design further enhances its resistance to overfitting [12], a common challenge in deep learning applications, thereby ensuring better generalization to unseen data.

Previous studies on medical image analysis using deep learning have often relied on transfer learning from large pretrained models [13], which can be computationally intensive and may not always be optimally tuned for specific characteristics of medical images. Although effective, such approaches can obscure the direct architectural impact on the task. Furthermore, some previous works may not explicitly address the rapid overfitting that can occur with smaller medical datasets. This study overcomes these limitations by developing a simplified custom CNN architecture from scratch, allowing focused investigation into its effectiveness with potentially small datasets. The explicit inclusion of several dropout layers directly addresses the overfitting challenge, ensuring model robustness and generalization, even with limited training data. The CNN architecture comprises 10 layers, structured sequentially from feature extraction to classification. The model is built from scratch (without using pretrained weights) in a lightweight approach, incorporating dropout at three key points to prevent overfitting.

II. The Proposed Method & Algorithm

The CNN architecture employed in this study consists of 10 layers, sequentially organized from feature extraction to classification. The model is custom-built (without using any pretrained weights) with a lightweight design, incorporating dropout at three critical points to mitigate overfitting. The detailed architecture is presented in Table 1.

Table 1. Custom CNN Model Architecture Layers

Layer Type	Parameters	Output	Aktivasi
Conv2D	filters=32, kernel_size=(3,3)	(222, 222, 32)	ReLU
MaxPooling2D	default (pool_size=(2,2))	(111, 111, 32)	-
Dropout	rate=0.25	(111, 111, 32)	-
Conv2D	filters=64, kernel_size=(3,3)	(109, 109, 64)	ReLU
MaxPooling2D	default (pool_size=(2,2))	(54, 54, 64)	-
Dropout	rate=0.25	(54, 54, 64)	-
Flatten	-	(186624,)	-
Dense	units=128	(128,)	ReLU
Dropout	rate=0.5	(128,)	-
Dense	units=2	(2,)	Softmax

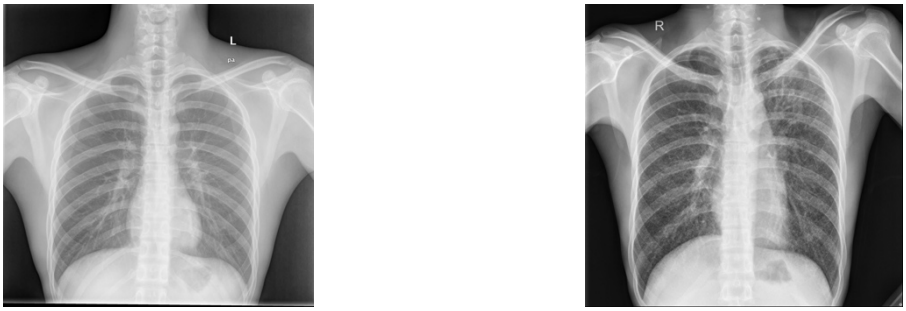
Table 1 illustrates the architecture of the custom CNN model designed for classifying chest X-ray images into two categories: normal and tuberculosis. The model begins with a Conv2D layer that extracts initial features from the input images using 32 filters of size 3×3, followed by a MaxPooling2D layer that reduces the spatial dimensions to accelerate computation and prevent

overfitting. A Dropout layer is applied after each convolutional block to reduce the risk of overfitting by randomly deactivating a subset of neurons during training. Next, a second Conv2D layer with 64 filters captures more complex features from the previously extracted representations, followed by another max-pooling and dropout layer. Once the feature extraction process is complete, a Flatten layer converts the 3D output into a 1D vector, which is then passed to a Dense layer with 128 neurons for initial classification. A third Dropout layer with a rate of 0.5 is applied before the output layer as an additional regularization step. Finally, a Dense layer with 2 units and a Softmax activation function is used to generate the class probabilities for normal and tuberculosis. The entire architecture is designed to be efficient and lightweight, making it suitable for small datasets and systems with limited computational resources.

III. Method

A. Dataset

The dataset used in this study is the Shenzhen TB dataset, which contains 662 chest X-ray images (336 TB and 326 normal), provided by the NIH and accompanied by labels, making it suitable for training and evaluating TB classification models [14]. The dataset was split into two subsets using the Keras ImageDataGenerator, with 80% allocated for training and 20% for validation.



(a) (b)

Fig. 1. X-ray images Shenzhen of (a) normal and (b) tuberculosis

Fig.1 presents two representative examples of chest X-ray images from the Shenzhen dataset, obtained from the Kaggle Tuberculosis Chest X-rays - Shenzhen. Fig1 (a) illustrates a normal lung condition, where the lung structures appear clear, symmetrical, and free from abnormal opacities. In contrast, Fig.1 (b) depicts the lungs of a tuberculosis patient, characterized by the presence of hazy areas (infiltrates) in several regions of the lung, indicating signs of active infection. This dataset is widely used in deep learning-based pulmonary disease classification research due to its high-resolution image quality and clinically verified labels.

B. Training and Evaluation

The training process was conducted using several hyperparameter settings. Model performance was evaluated using accuracy, loss, confusion matrix, and classification report metrics.

Table 2. Hyperparameters setting

Hyperparameters	Output
Epoch	10, 25, 50
Optimizer	Adam
Batch size	32
Loss Function	Categorical Crossentropy

The number of epochs tested included 10, 25, and 50, aimed at observing the effect of training duration on the model's accuracy. The Adam optimizer was employed due to its effectiveness and adaptability in dynamically updating model weights. A batch size of 32 was selected as a trade-off between memory efficiency and training stability. The loss function used was Categorical Crossentropy, which is appropriate for multi-class classification involving two categories: TB and

normal. These parameter choices were made to ensure an efficient model that converges quickly while maintaining high accuracy [15].

These evaluation metrics are derived from the confusion matrix, which records the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). By utilizing these metrics, this study aims to achieve a robust and comprehensive performance assessment [16].

Accuracy assesses the overall correctness of the model's predictions by calculating the ratio of correctly predicted instances to the total number of samples, it is defined as Eq.1 :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision measures the proportion of positive predictions that are truly correct, providing insight into the model's reliability in identifying the positive class. Precision can be computed by Eq.2:

$$precision = \frac{TP}{TP+FP} \quad (2)$$

Recall quantifies the model's ability to correctly identify all actual positive cases. Recall is calculated using Eq.3:

$$recall = \frac{TP}{TP+FN} \quad (3)$$

F1-score represents the harmonic mean of precision and recall, offering a balanced metric particularly useful in cases of class imbalance. F1-scores formulated as Eq.4:

$$F1 - score = \frac{2 \times Recall \times Precision}{Recall+Precision} \quad (4)$$

IV. Results and Discussion

This section presents the evaluation results of the custom CNN model trained using the defined parameters. The evaluation is based on accuracy, loss, confusion matrix, and classification report metrics to assess the model's performance in classifying X-ray images into two classes: normal and tuberculosis. Furthermore, an analysis is conducted on the impact of the number of training epochs on validation accuracy, aiming to determine the point at which the model achieves optimal performance without overfitting. The results are analyzed thoroughly to understand the strengths, limitations, and potential future enhancements of the proposed model.

A. Training Result

The training process produced historical data of accuracy and loss values for both training and validation data, visualized through graphs. These visualizations provide an overall view of the model's learning stability and effectiveness under each epoch scenario and help identify the optimal training point based on validation performance. The graphs below present the training and validation accuracy and loss for each experiment.

Fig.2 presents the accuracy and loss curves obtained during the training process of the CNN model under three different epoch scenarios: (a) 10 epochs, (b) 25 epochs, and (c) 50 epochs. Each subgraph consists of two parts: the left graph illustrates the progression of training accuracy (Train Acc) and validation accuracy (Val Acc), while the right graph depicts the loss values on both training and validation data throughout the training process.

Fig.2 (a), both training and validation accuracy increase steadily, reaching a peak of approximately 0.86 at the 10th epoch. Meanwhile, the loss decreases sharply in the early stages and then stabilizes, indicating that the model learns effectively without signs of overfitting. Fig.2 (b) shows the training results for 25 epochs, where training accuracy continues to improve, but validation accuracy tends to plateau after the 10th epoch. This suggests that the model begins to exhibit symptoms of overfitting beyond this point, even though the loss continues to decrease gradually.

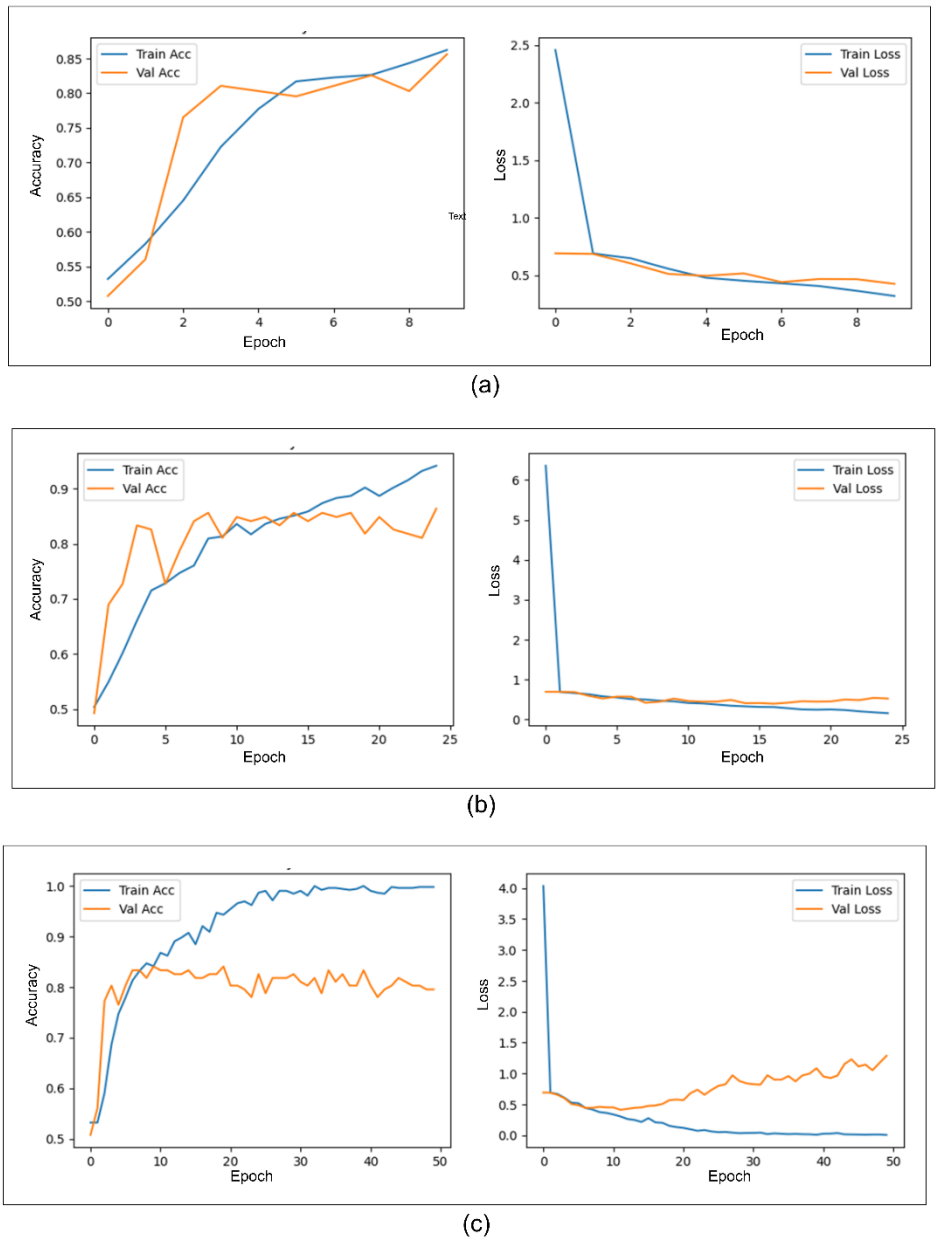


Fig. 2. Training accuracy and loss graph epoch (a) 10, (b) 25, (c) 50

Figure 2 (c), which represents training over 50 epochs, the overfitting trend becomes more apparent. Training accuracy approaches 1.0, while validation accuracy shows no significant improvement and even begins to decline. Additionally, the validation loss increases after the 20th epoch, further indicating overfitting.

Overall, the plots in Figure 2 demonstrate that training the model for 10 epochs yields the best balance between training and validation accuracy, as well as stable loss values. In contrast, extended training leads to overfitting and a decline in validation performance. These findings reinforce the observation that increasing the number of epochs beyond a certain point does not necessarily improve model performance, particularly when working with a limited dataset.

B. Evaluation

Figure 3 illustrates the confusion matrices obtained from the CNN model under three different training durations: (a) 10 epochs, (b) 25 epochs, and (c) 50 epochs. Each matrix presents the number of correct and incorrect predictions for the two classes: NORMAL and TB. These visualizations

provide a detailed breakdown of the model's classification performance, offering insight into the distribution of true positives, true negatives, false positives, and false negatives for each training scenario.

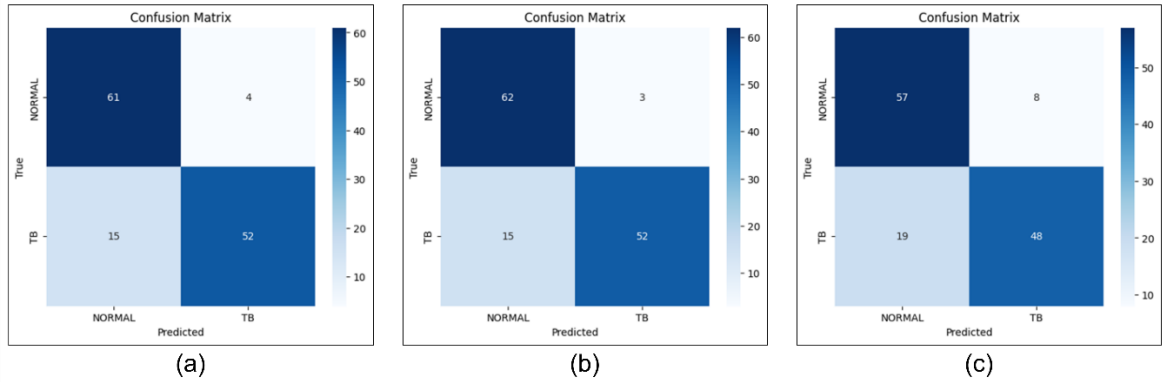


Fig. 3. Confusion Matrix (a) 10 epoch, (b) 25 epoch, (c) 50 epoch

In Figure 3 (a), the model trained for 10 epochs correctly classified 61 out of 65 NORMAL images and 52 out of 67 TB images. There were 4 misclassifications in the NORMAL class and 15 in the TB class. This indicates that the model performs in a relatively balanced manner across both classes with stable classification accuracy. Fig.3 (b) shows the results after 25 epochs of training. A slight improvement is observed in the NORMAL class with 62 correct and only 3 incorrect predictions. However, the classification performance for the TB class remains unchanged, with 52 correct and 15 incorrect predictions. This suggests that while a minor improvement occurred in one class, there was no significant enhancement in overall performance. Conversely, Fig.3(c), which depicts the model trained for 50 epochs, shows a decline in performance. The number of correct classifications for NORMAL dropped to 57, and for TB to 48, while misclassifications increased in both classes 8 for NORMAL and 19 for TB indicating a deterioration likely caused by overfitting.

Table 3. Comparison Evaluation

Epoch	Accuracy	Precision	Recall	F1-score
10	85.61%	92.86%	77.61%	84.27%
25	86.36%	94.55%	77.61%	85.23%
50	79.55%	85.71%	71.64%	78.02%

Table 3 presents the performance evaluation of the custom Convolutional Neural Network (CNN) model based on four key binary classification metrics: accuracy, precision, recall, and F1-score. These metrics were computed from the confusion matrix for each training scenario: 10, 25, and 50 epochs. The model trained for 25 epochs achieved the highest accuracy (86.36%) and precision (94.55%), demonstrating its capability to accurately detect tuberculosis cases. However, this performance represents only a slight improvement compared to the 10-epoch model, which achieved an accuracy of 85.61% and precision of 92.86%, while requiring significantly less training time and computational resources. Additionally, the 10-epoch model produced a competitive F1-score of 84.27%, reflecting a well-balanced trade-off between precision and recall.

In contrast, the model trained for 50 epochs exhibited performance degradation across all metrics, most notably in recall (71.64%) and accuracy (79.55%). This decline strongly indicates the presence of overfitting where the model becomes excessively fitted to the training data, resulting in diminished generalization capability on unseen data. The performance drop is also evidenced by an increase in both false negatives and false positives, particularly in the tuberculosis class, signaling the model's reduced effectiveness in identifying TB cases.

Overall, these findings suggest that a training duration between 10 and 25 epochs is optimal, especially in the context of a small-scale dataset such as the Shenzhen chest X-ray set used in this study. This range provides a balance between precision and generalization while minimizing the risk of overfitting, which is commonly observed in CNN models trained for extended epochs without

appropriate regularization mechanisms, such as early stopping or learning rate decay. These results are consistent with prior studies indicating that deep learning models trained on limited datasets are highly susceptible to overfitting unless the number of epochs and model complexity are carefully controlled.

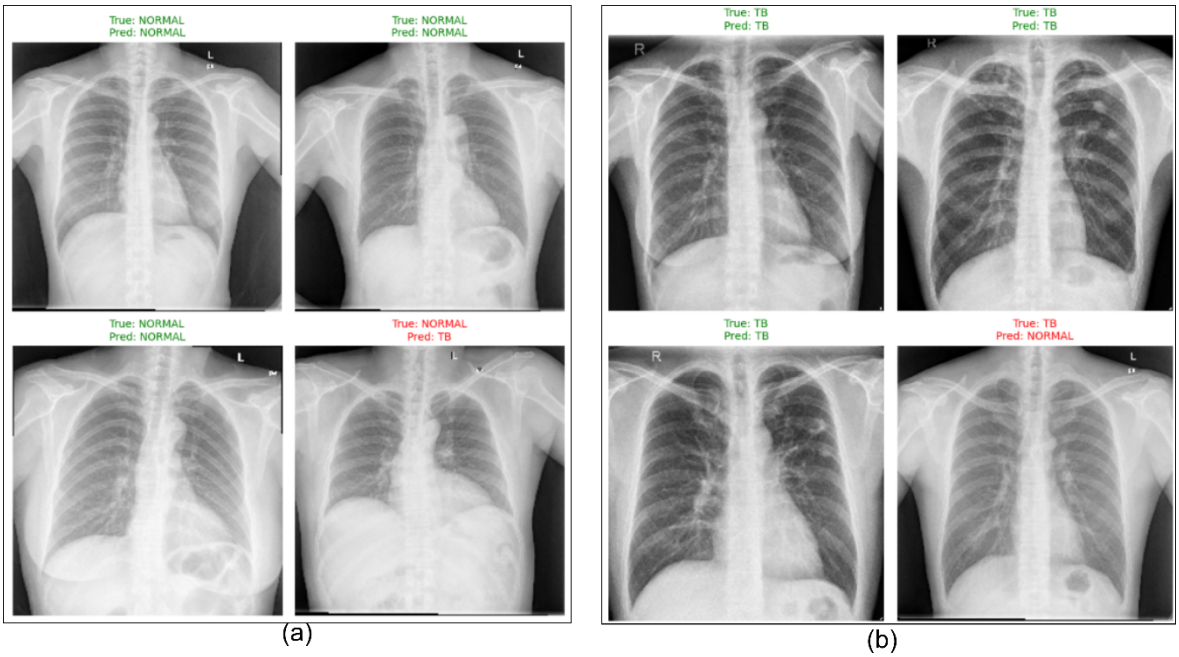


Fig. 4. Visual examples of CNN model classifications (a) Normal and (b) TB

Figure 4 presents representative visual examples of CNN model classifications for chest X-ray images from the two classes: (a) NORMAL and (b) TB. Each subfigure consists of four images labeled according to their ground truth (True) and the model's prediction (Pred). Green highlights indicate correct predictions, while red indicates misclassifications. In part (a), three out of four NORMAL images were correctly classified, with one image misclassified as TB. A similar pattern is observed in part (b), where three out of four TB images were accurately identified, while one TB image was incorrectly classified as NORMAL. This visualization reinforces the understanding that, despite the model's overall high performance, some misclassifications still occur potentially due to similarities in radiological features between the two classes or variations in image quality.

V. Conclusion

This study demonstrates that a lightweight and efficient custom Convolutional Neural Network (CNN) architecture can be effectively used to classify chest X-ray images into two categories: normal and tuberculosis. With a simple model configuration and the use of dropout as a regularization technique, the model was able to achieve optimal performance within just 10 to 25 training epochs. Evaluation results show that training for 10 epochs yielded an accuracy of 85.61%, precision of 92.86%, and an F1-score of 84.27%, which was comparable to, and in some aspects more stable than, the results obtained with 25 epochs. In contrast, training the model for 50 epochs led to a decline in performance due to overfitting, as reflected in the reduced accuracy and recall scores.

These findings confirm that extended training does not necessarily lead to improved results, especially when working with small datasets such as the Shenzhen chest X-ray dataset. Therefore, the use of a custom CNN with a short training duration offers an efficient and practical solution for automated TB detection systems, particularly in deployment scenarios involving limited computational resources. Future research is encouraged to incorporate advanced data augmentation techniques and to compare this custom architecture with pretrained models such as MobileNet, EfficientNet, or Vision Transformer

References

[1] WHO, *Tuberculosis Report*, vol. XLIX, no. 9-10–11. 2020. [Online]. Available: https://iris.who.int/handle/10665/336069

[2] A. Iqbal, M. Usman, and Z. Ahmed, "Tuberculosis chest X-ray detection using CNN-based hybrid segmentation and classification approach," *Biomed. Signal Process. Control*, vol. 84, p. 104667, 2023, doi: https://doi.org/10.1016/j.bspc.2023.104667.

[3] U. Ahmad *et al.*, "Large Scale Fish Images Classification and Localization using Transfer Learning and Localization Aware CNN Architecture," *Comput. Syst. Sci. Eng.*, vol. 45, no. 2, pp. 2125–2140, 2023, doi: 10.32604/csse.2023.031008.

[4] J. Ao *et al.*, "Stimulated Raman Scattering Microscopy Enables Gleason Scoring of Prostate Core Needle Biopsy by a Convolutional Neural Network," *Cancer Res.*, vol. 83, no. 4, pp. 641–651, 2023, doi: 10.1158/0008-5472.CAN-22-2146.

[5] K. Murphy *et al.*, "Computer aided detection of tuberculosis on chest radiographs: An evaluation of the CAD4TB v6 system," *Sci. Rep.*, vol. 10, no. 1, pp. 1–11, 2020, doi: 10.1038/s41598-020-62148-y.

[6] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "DeTrac: Transfer Learning of Class Decomposed Medical Images in Convolutional Neural Networks," *IEEE Access*, vol. 8, pp. 74901–74913, 2020, doi: 10.1109/ACCESS.2020.2989273.

[7] W. Ren, A. Hasanzade Bashkandi, J. Afshar Jahanshahi, A. Qasim Mohammad AlHamad, D. Javaheri, and M. Mohammadi, "Brain tumor diagnosis using a step-by-step methodology based on courtship learning-based water strider algorithm," *Biomed. Signal Process. Control*, vol. 83, p. 104614, 2023, doi: https://doi.org/10.1016/j.bspc.2023.104614.

[8] M. K. Puttagunta and S. Ravi, "Detection of Tuberculosis based on Deep Learning based methods," *J. Phys. Conf. Ser.*, vol. 1767, no. 1, 2021, doi: 10.1088/1742-6596/1767/1/012004.

[9] K. Jin *et al.*, "MSHF: A Multi-Source Heterogeneous Fundus (MSHF) Dataset for Image Quality Assessment," *Sci. Data*, vol. 10, no. 1, pp. 1–6, 2023, doi: 10.1038/s41597-023-02188-x.

[10] J. Escorcia-Gutierrez, M. Gamarra, R. Soto-Diaz, S. Alsafari, A. Yafoz, and R. F. Mansour, "Optimal Synergic Deep Learning for COVID-19 Classification Using Chest X-Ray Images," *Comput. Mater. Contin.*, vol. 75, no. 3, pp. 5255–5270, 2023, doi: 10.32604/cmc.2023.033731.

[11]A. Wong, J. R. H. Lee, H. Rahmat-Khah, A. Sabri, A. Alaref, and H. Liu, "TB-Net: A Tailored, Self-Attention Deep Convolutional Neural Network Design for Detection of Tuberculosis Cases From Chest X-Ray Images," *Front. Artif. Intell.*, vol. 5, no. April, pp. 1–10, 2022, doi: 10.3389/frai.2022.827299.

[12]S. A. Salloum, K. M. Alomari, A. M. Alfaisal, R. A. Aljanada, and A. Basiouni, "Emotion recognition for enhanced learning: using AI to detect students' emotions and adjust teaching methods," *Smart Learn. Environ.*, vol. 12, no. 1, 2025, doi: 10.1186/s40561-025-00374-5.

[13] M. Irhamsyah, Q. A'yuni, K. Saddami, N. Nasaruddin, K. Munadi, and F. Arnia, "Impact of Using Various X-Ray Dataset in Detecting Tuberculosis Based on Deep Learning," *Radioelectron. Comput. Syst.*, vol. 2025, no. 1(113), pp. 165–186, 2025, doi: 10.32620/reks.2025.1.12.

[14] S. Jaeger, S. Candemir, S. Antani, Y.-X. J. Wáng, P.-X. Lu, and G. Thoma, "Two public chest X-ray datasets for computer-aided screening of pulmonary diseases.," *Quant. Imaging Med. Surg.*, vol. 4, no. 6, pp. 475–477, Dec. 2014, doi: 10.3978/j.issn.2223-4292.2014.11.20.

[15] Q. A'yuni, N. Nasaruddin, M. Irhamsyah, M. Azhary, and R. Roslidar, "Intelligent Tuberculosis Detection System with Continuous Learning on X-ray Images," *J. Electron. Electromed. Eng. Med. Informatics*, vol. 7, no. 1, pp. 130–141, 2025, doi: 10.35882/jeeemi.v7i1.572.

[16] A. Ahmadiar, M. Melinda, Z. Muthiah, Z. Zainal, and M. M. Rizky, "Thermal Image Classification of Autistic Children Using Res-Net Architecture," vol. 7, no. 1, pp. 1–10, 2025.