



Attention-augmented cnn for lung disease detection: integrating cbam with transfer learning architectures

Tran Van Thien^{1*}, Ngo Van Den¹, Le Vo Hoang Duy¹, Tran Thanh Nam¹, and Nguyen Van Linh¹

¹College of Digital Technology and Artificial Intelligence, Nam Can Tho University

*Corresponding author: tranvanthien@nctu.edu.vn

DOI: 10.64632/jsde.eng.4.2026.877

ABSTRACT

Received: Apr 19, 2026

Revised: Apr 23, 2026

Accepted: May 15, 2026

Keywords: CBAM, CNN, deep learning, EfficientNetB0, lung disease diagnosis, MobileNetV2, ResNet50

This study focuses on developing and comparing five deep learning models (CNN with CBAM, CNN with TFRecord and Otsu thresholding, MobileNetV2, ResNet50, and EfficientNetB0) for classifying five lung conditions (normal, bacterial pneumonia, viral pneumonia, tuberculosis, and COVID-19) from chest X-ray images, aiming to support healthcare in underserved regions of Vietnam. Experimental results showed that the ResNet50 model achieved the highest performance with a test accuracy of 90.38%, outperforming CNN with CBAM (87.80%), CNN with grayscale preprocessing (77.16%), MobileNetV2 (89.98%), and EfficientNetB0 (88.59%). The models were trained on a dataset of 10,000 X-ray images from Kaggle 2022, carefully processed with data augmentation techniques, particularly for bacterial and viral pneumonia. The system was integrated into a Flask-based web application with a Vietnamese interface, enabling rapid image uploads, real-time feedback via WebSocket, and storage of diagnostic history in JSON format. The study provides a feasible technical solution with significant potential for practical application in community healthcare in Vietnam.

1. INTRODUCTION

Lung-related diseases, including tuberculosis, bacterial pneumonia, viral pneumonia, and COVID-19, continue to pose a significant global health burden. According to the World Health Organization, respiratory diseases account for

more than three million deaths annually, remaining among the leading causes of mortality worldwide (W.H.O., 2024). In developing countries such as Vietnam, this burden is further exacerbated by disparities in healthcare infrastructure, particularly in rural and remote

regions. Limited access to advanced diagnostic equipment, such as high-resolution chest X-ray systems and experienced radiologists, significantly constrains early detection and timely treatment. Conventional diagnostic practices, which rely heavily on manual interpretation of radiographic images, are inherently subjective and dependent on clinician expertise, often resulting in variability in diagnostic accuracy especially during large-scale outbreaks when healthcare systems are overwhelmed.

In this context, artificial intelligence (AI), particularly deep learning, has emerged as a transformative approach for medical image analysis. Deep learning models have demonstrated remarkable capability in automatically extracting hierarchical features from complex imaging data, thereby improving both the speed and accuracy of diagnosis (Litjens et al., 2017). The application of AI in radiology has shown promising results in assisting clinicians with disease detection, reducing diagnostic workload, and enhancing decision-making processes, particularly in resource-constrained environments (He et al., 2016).

This study proposes a comprehensive framework for automated lung disease classification using chest X-ray images. Specifically, we develop and compare five deep learning models, including CNN integrated with Convolutional Block Attention Module (CBAM), CNN combined with TFRecord and Otsu thresholding, MobileNetV2, ResNet50, and EfficientNetB0 (Krizhevsky et al., 2012; Simonyan & Zisserman, 2015; Tan & Le, 2020). All models leverage transfer learning to enhance performance on limited datasets and are

optimized for chest X-ray data relevant to the Vietnamese context (Mahamud et al., 2024). Beyond model development, this research emphasizes practical applicability by integrating the trained models into a Flask-based web application. The system features a user-friendly Vietnamese interface, rapid image processing capabilities, and diagnostic history storage in JSON format, facilitating longitudinal tracking and usability in real-world scenarios. The overall research workflow encompasses data acquisition and preprocessing, model design and training on cloud platforms such as Google Colab and Kaggle, performance evaluation, and the proposal of optimization strategies.

The evolution of AI in healthcare has progressed through several stages. Early approaches predominantly utilized traditional machine learning algorithms such as Support Vector Machine (SVM) and Random Forests, which relied on handcrafted features and exhibited limited performance when applied to high-dimensional image data (Breiman, 2001; Chandra & Bedi, 2021; Genauer & Poggi, 2020). The introduction of Convolutional Neural Networks (CNNs), notably AlexNet, marked a significant breakthrough in image classification tasks (Krizhevsky et al., 2012). Subsequent architectures, including VGGNet (Simonyan & Zisserman, 2015), ResNet (He et al., 2016), MobileNetV2 (Sandler et al., 2018), and EfficientNet (Tan & Le, 2019), have further advanced the field by improving both accuracy and computational efficiency. These models have been successfully applied in numerous studies for detecting lung cancer, classifying pneumonia, and supporting COVID-19 diagnosis, primarily in data-rich and technologically advanced

settings (Mahamud et al., 2024; Apostolopoulos & Mpesiana, 2020). Recent studies have increasingly leveraged transfer learning to address the limitation of labeled medical datasets, significantly improving model generalization and reducing training time. Data augmentation techniques have also been widely adopted to mitigate overfitting and enhance robustness (Rajpurkar et al., 2017). Additionally, attention mechanisms such as Convolutional Block Attention Module (CBAM) have been integrated into CNN architectures to improve feature representation, particularly in detecting subtle abnormalities in chest X-ray images (Woo et al., 2018).

Despite these advancements, most existing works are conducted using large-scale datasets from developed countries, which may not generalize well to local contexts such as Vietnam. Furthermore, many studies prioritize model accuracy without adequately considering deployment constraints, such as limited computational resources, lack of standardized datasets, and the need for user-friendly clinical applications. Practical implementation in real-world healthcare systems especially in rural or underserved regions remains limited. Therefore, a significant research gap exists in developing lightweight, accurate, and deployable deep learning models tailored to local datasets and real-world conditions. This study addresses this gap by systematically comparing multiple architectures, including CNN variants with attention mechanisms and transfer learning models, while also integrating the optimal model into a web-based diagnostic system to enhance practical applicability in resource-constrained healthcare settings.

2. MATERIALS AND METHODS

2.1 Dataset

This study utilizes the Lung Disease dataset (4 types), published by Omkar Manohar Dalvi on Kaggle in 2022 (Dalvi, 2022). The dataset comprises chest X-ray images collected from various reputable medical sources, processed using VisiPics to remove duplicates, ensuring data uniqueness. According to Kaggle, the initial dataset was augmented by a factor of 6, targeting approximately 10,000 images. However, after verification through programmatic testing, the actual total reached 10,095 samples, allocated as follows: 6,054 training samples, 2,016 validation samples, and 2,025 test samples. This difference may stem from the data augmentation process or minor errors in the initial reporting but does not significantly affect the results. The inconsistency in image size, with varying resolutions (e.g., 224×224, 256×256, or 512×512 pixels), necessitates appropriate preprocessing steps. The images are primarily in grayscale format, although some have different color channels with 8-bit depth, and the most common resolution is 224×224 pixels, suitable for training deep learning models.

The dataset is categorized into five lung pathology groups: Normal, Bacterial Pneumonia, Viral Pneumonia, Coronavirus Disease (COVID-19), and Tuberculosis. The data are organized into three folders: training, validation, and testing, each containing subfolders corresponding to the respective pathology groups. The training dataset, with the largest sample size, is used to train the model; the validation dataset supports hyperparameter refinement; and the test dataset is used for independent evaluation. The dataset is highly reliable, sourced from reputable

healthcare organizations, and has been used in numerous studies. The diversity of lung pathologies represented in the dataset reflects common real-world cases, making it highly

suitable for the diagnosis of lung diseases. Furthermore, its accessibility and ease of downloading from Kaggle facilitate efficient integration into the model training process.



**Figure 1 Nor-
mal**



**Figure 2 Bacte-
rial Pneumonia**



**Figure 3 Viral
Pneumonia**



**Figure 4
COVID-19**



**Figure 5
Tuberculosis**

2.2 Data preprocessing

The data preprocessing process plays a crucial role in preparing X-ray images for five deep learning models (CNN_v0, CNN_v1, MobileNetV2, ResNet50, and EfficientNetB0). Due to image size inconsistencies in the dataset, the preprocessing steps are designed to normalize and optimize the input data.

Detailed preprocessing steps include: Images are read and validated using `tf.io.read_file` and `tf.image.decode_image`, supporting multiple formats (.jpg, .jpeg, .png, .bmp, .gif, .tiff, or .tif) with the number of channels set to 3 (RGB) and animation exclusion (`expand_animations=False`). Image integrity is ensured by discarding corrupted files, identified by the output `None`. To address size inconsistencies, all images were resized to 224×224 pixels using `tf.image.resize`, matching the input requirements of models such as ResNet50 and EfficientNetB0. For the custom CNN_v1 model, grayscale images were processed separately using `tf.image.rgb_to_grayscale` to preserve grayscale formatting when necessary, then normalized pixel values to the range $[0, 1]$ by dividing by 255, instead of converting to RGB as for other models. Grayscale images not converted to RGB

underwent additional normalization using specific mean and standard deviation values (mean = 0.5, standard deviation = 0.5) to match the CNN_v1 architecture.

Grayscale images are converted to RGB by copying pixel values across three channels as needed for models such as MobileNetV2 and EfficientNetB0, using `cv2.cvtColor(img, cv2.COLOR_GRAY2RGB)` or `tf.image.decode_image` with `channels=IMG_CHANNELS`. Pixel values are normalized using `applications.resnet50.preprocess_input` or methods compatible with MobileNetV2 and EfficientNetB0, based on the mean and standard deviation of the ImageNet dataset.

To minimize potential noise, input data was preprocessed from the original dataset, although no direct noise reduction filters were applied in the code. In the future, Gaussian filters may be considered to improve image quality, especially for images with inconsistent resolution. To address the data imbalance, data enhancement techniques were applied to the Bacterial Pneumonia and Viral Pneumonia layers (labels 1 and 4) using `tf.keras.Sequential`, which included random transformations: horizontal flip

(layers.RandomFlip("horizontal")), random rotation within ± 0.2 radians (approximately ± 11.5 degrees) (layers.RandomRotation(0,2)), random zoom within $\pm 10\%$ (layers.RandomZoom(0,1)), and random contrast adjustment within $\pm 20\%$ (layers.RandomContrast(0,2)). This process is conditionally applied using tf.cond to target only these two layers, with the output converted to tf.float16 to optimize memory usage, supporting the performance of all five models, including the custom model CNN_v1.

The data path is constructed using tf.data.Dataset with a batch size of

BATCH_SIZE=32, using parallel preprocessing with `num_parallel_calls=tf.data.AUTOTUNE` to improve performance. For the training set, data is shuffled with `buffer_size=len(image_paths)` when shuffle mode is enabled, ensuring randomness. The data is set up to repeat (repeat()) and prefetch (prefetch(tf.data.AUTOTUNE)) to maintain a continuous data flow, suitable for the training requirements of models on platforms such as Kaggle. This path ensures stability and efficiency during the training of custom CNN_v0, CNN_v1, MobileNetV2, ResNet50, and EfficientNetB0 models.

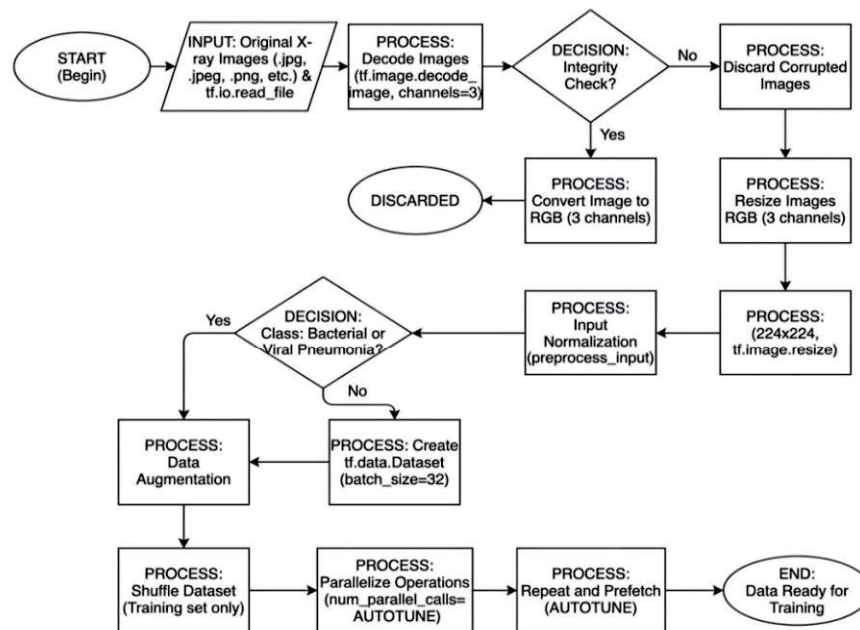


Fig. 6. Flowchart of the data preprocessing pipeline

(Source: Author's diagram)

3. RESULTS AND DISCUSSION

3.1 Experimental setup

The models were initially trained on Google Colab and subsequently migrated to Kaggle to leverage free GPU resources, ensuring high computational efficiency. The training pipeline was designed with 100 epochs for most models

(cnn_v0_rgb, mobilenetv2_v1_rgb, resnet50_v1_rgb, efficientnetb0_v1_rgb), except for cnn_v1_tfrecord_grayscale_otsu, which used only 23 epochs due to early convergence. All models employed the Adam optimizer with an initial learning rate of 0.0001, reduced by 10% every 10 epochs via a scheduler to optimize

convergence. The loss function was categorical cross-entropy, combined with a dropout rate of 0.3 to mitigate overfitting. The batch size was set to 32, adjusted from 64 due to server limitations on Kaggle, which impacted training time but not model quality. Specifically, the `cnn_v1_tfrecord_grayscale_otu` model utilized cosine decay for flexible learning rate adjustment. Fine-tuning focused on the last 30 layers for MobileNetV2 and ResNet50, and 50 layers for EfficientNetB0, leveraging pre-trained architectures from the ImageNet dataset.

3.2 Training results

The models were evaluated based on loss and accuracy metrics on the training, validation, and test sets from the Lungs Disease Dataset (Dalvi, 2022). The `cnn_v0_rgb` model achieved a Train Loss of 0.2735, Train Accuracy of 90.13%, Validation Loss of 0.3630, Validation Accuracy of 87.99%, and Test Accuracy of 87.80% after 50 epochs, demonstrating that CBAM integration enhances focus on critical features, although performance is limited by the lack of architectural depth.

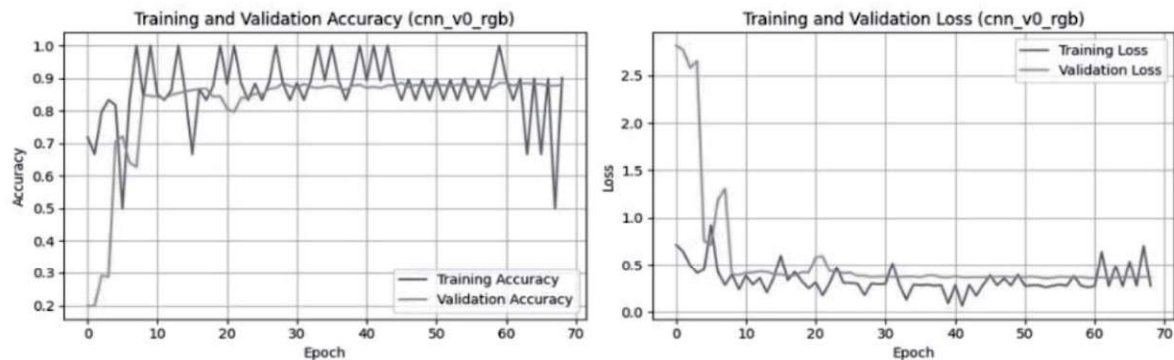


Fig. 7. Plot of loss and accuracy for `cnn_v0_rgb`

(Source: Author's diagram)

Meanwhile, the `cnn_v1_tfrecord_grayscale_otu` model achieved a Train Loss of 1.0997, Train Accuracy of 49.39%, Validation Loss of 0.5615, Validation Accuracy of 76.55%, and Test Accuracy of 77.16% after 23

epochs, utilizing Otsu thresholding and the TFRecord format to optimize data processing. However, the use of grayscale images reduces color information, thereby impacting overall performance.

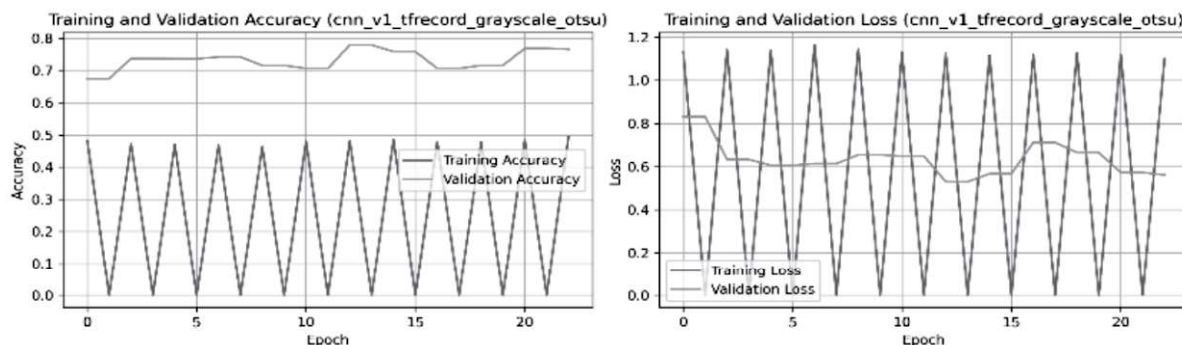


Fig. 8. Plot of loss and accuracy for `cnn_v1_tfrecord_grayscale_otu`

(Source: Author's diagram)

The mobilenetv2_v1_rgb model achieved a Train Loss of 0.2650, Train Accuracy of 90.22%, Validation Loss of 0.2940, Validation Accuracy of 89.29%, and Test Accuracy of 89.98% after

100 epochs, demonstrating high performance and suitability for resource-constrained devices due to its lightweight design.

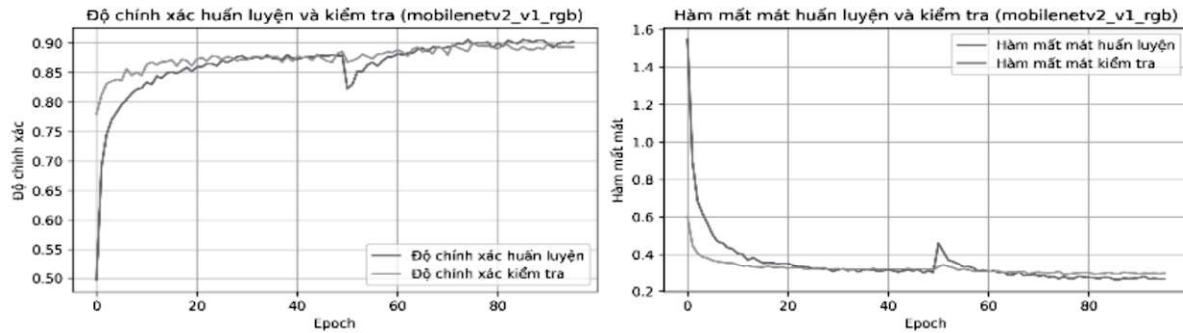


Fig. 9. Plot of loss and accuracy for mobilenetv2_v1_rgb

(Source: Author's diagram)

The ResNet50_v1_RGB model demonstrates outstanding performance, achieving a Train Loss of 0.2216, Train Accuracy of 93.13%, Validation Loss of 0.2954, Validation Accuracy of 89.63%,

and Test Accuracy of 90.38% after 100 epochs. The residual blocks significantly contribute to the model's superior performance, particularly on the test set.

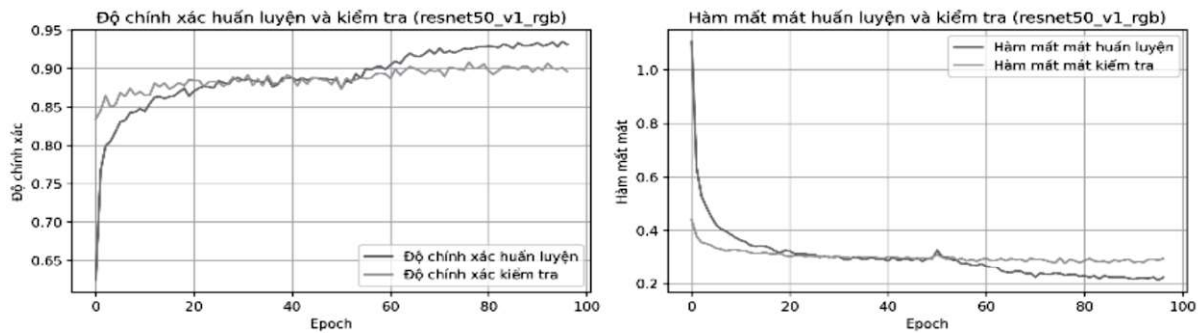


Fig. 10. Plot of loss and accuracy for resnet50_v1_rgb

(Source: Author's diagram)

Ultimately, the EfficientNetB0_v1_RGB model achieved a Train Loss of 0.3733, Train Accuracy of 87.31%, Validation Loss of 0.3444, Validation Accuracy of 88.84%, and Test

Accuracy of 88.59% after 100 epochs. The model maintains stability due to compound scaling, although its performance is lower than that of ResNet50.

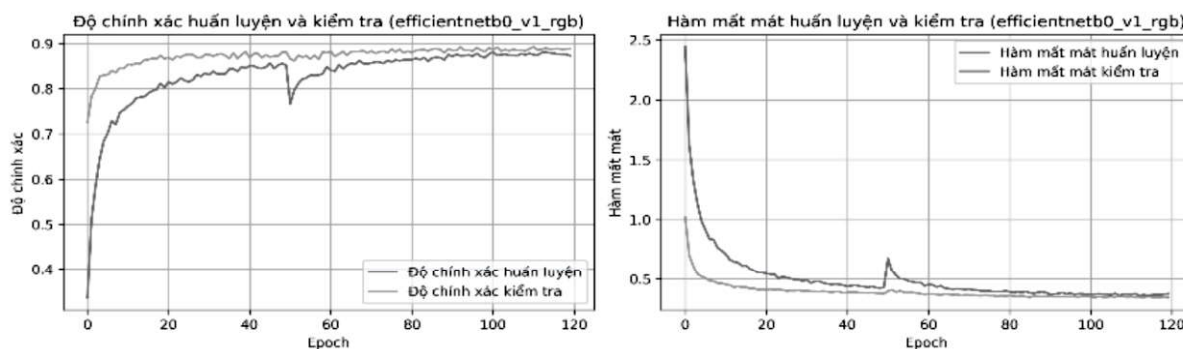


Fig. 11. Plot of loss and accuracy for efficientnetb0_v1_rgb

(Source: Author's diagram)

3.3 Evaluation results

The confusion matrix and classification report were constructed to evaluate the classification performance of each model on the test set.

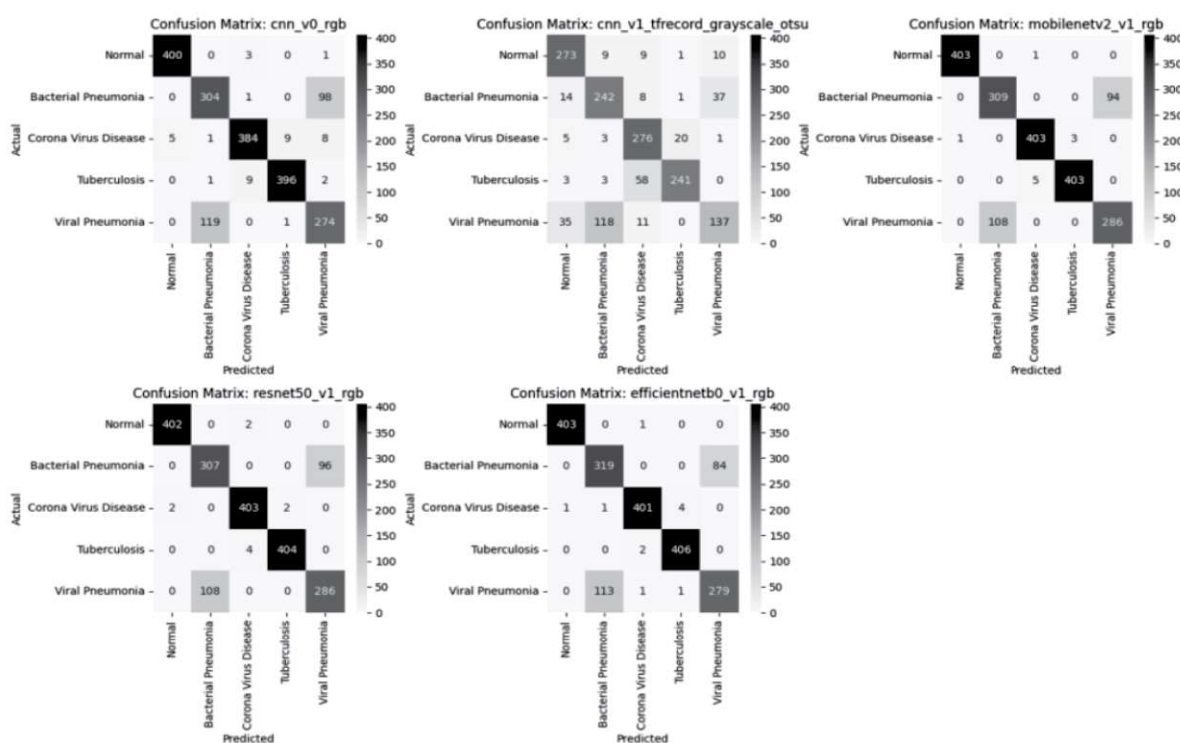


Fig. 12. Confusion matrix of the models

(Source: Author's diagram)

The `cnn_v0_rgb` model demonstrated a classification accuracy of 400/404 samples for the Normal group and 384/407 for the COVID-19 group. However, it achieved lower accuracies of 304/403 for Bacterial Pneumonia and 274/394 for Viral Pneumonia, indicating confusion

between the pneumonia groups due to their similar characteristics.

The `cnn_v1_tfrecord_grayscale_otsu` model achieved classification accuracies of 273/302 for the Normal group, 276/305 for COVID-19, 242/302 for Bacterial Pneumonia, and 137/301

for Viral Pneumonia, exhibiting lower performance attributed to the absence of color information.

The mobilenetv2_v1_rgb model achieved classification accuracies of 403/404 for the Normal group, 403/407 for COVID-19, 309/403 for Bacterial Pneumonia, and 286/394 for Viral Pneumonia, demonstrating high accuracy.

The ResNet50_v1_RGB model achieved classification accuracies of 402/404 for the Normal group, 403/407 for COVID-19, 307/403 for Bacterial Pneumonia, and 286/394 for Viral Pneumonia, demonstrating stable performance.

The EfficientNetB0_v1_RGB model achieved classification accuracies of 403/404 for the Normal group, 401/407 for COVID-19, 319/403 for Bacterial Pneumonia, and 279/394 for Viral Pneumonia, indicating strong discriminative performance.

The classification report provides precision, recall, F1-score, and support metrics for each class, along with macro averages and weighted averages. Detailed data from the test set are as follows (percentages for precision, recall, and F1-score):

Table 1. Classification report of cnn_v0_rgb

Model	Class	Precision	Recall	F1- Score	Support
cnn_v0_rgb	Normal	98.77	99.01	98.89	404
	Bacterial Pneumonia	71.53	75.43	73.43	403
	Corona Virus Disease	96.73	94.35	95.52	407
	Tuberculosis	97.54	97.06	97.3	408
	Viral Pneumonia	71.54	69.54	70.53	394
	Macro Avg	87.22	87.08	87.13	2016
	Weighted Avg	82.73	87.2	87.25	2016

Table 2. Classification report of cnn_v1_tfrecord_grayscale_otu

Model	Class	Precision	Recall	F1- Score	Support
cnn_v1_tfrecord _grayscale_otu	Normal	82.73	90.4	86.39	302
	Bacterial Pneumonia	64.53	80.13	71.49	302
	Corona Virus Disease	76.24	90.49	82.76	305
	Tuberculosis	91.63	79.02	84.86	305
	Viral Pneumonia	74.05	45.52	56.38	301
	Macro Avg	77.84	77.11	76.38	1515
	Weighted Avg	77.87	77.16	76.42	1515

Table 3. Classification report of mobilenetv2_v1_rgb

Model	Class	Precision	Recall	F1- Score	Support
mobilenetv2_v1_rgb	Normal	99.75	99.75	99.75	404
	Bacterial Pneumonia	74.1	76.67	75.37	403
	Corona Virus Disease	98.53	99.02	98.77	407
	Tuberculosis	99.26	97.87	98.62	408
	Viral Pneumonia	75.26	72.59	73.9	394
	Macro Avg	89.38	89.38	89.28	2016
	Weighted Avg	89.49	89.48	89.48	2016

Table 4. Classification report of resnet50_v1_rgb

Model	Class	Precision	Recall	F1- Score	Support
resnet50_v1_rgb	Normal	99.50	99.50	99.50	404
	Bacterial Pneumonia	73.98	76.18	75.06	403
	Corona Virus Disease	98.53	99.02	98.77	407
	Tuberculosis	99.51	99.02	99.26	408
	Viral Pneumonia	74.87	72.59	73.71	394
	Macro Avg	89.28	89.26	89.26	2016
	Weighted Avg	89.39	89.38	89.38	2016

Table 5. Classification report of efficientnetb0_v1_rgb

Model	Class	Precision	Recall	F1- Score	Support
efficientnetb0_v1_rgb	Normal	99.75	99.75	99.75	404
	Bacterial Pneumonia	73.67	79.16	76.32	403
	Corona Virus Disease	99.01	98.53	98.77	407
	Tuberculosis	98.78	99.51	99.15	408
	Viral Pneumonia	76.86	70.81	73.71	394
	Macro Avg	89.62	89.55	89.54	2016
	Weighted Avg	89.72	89.68	89.66	2016

The model has been deployed and integrated into a web application, enabling rapid image analysis and access. Additionally, the system

incorporates a diagnostic history storage function, enhancing the efficiency of management and supporting medical decision-making.

**Fig. 13. System homepage**

(Source: Author's diagram)

4. CONCLUSION

This study presents an effective deep learning-based framework for automated lung disease classification from chest X-ray images,

combined with a practical web-based deployment. Among the evaluated models, the ResNet50_v1_rgb architecture achieved the best performance, with a test accuracy of 90.38% and

strong consistency across training and validation sets, demonstrating robust generalization capability. The integration of the trained model into a Flask-based application further validates the system's feasibility, enabling fast, user-friendly diagnosis and supporting functionalities such as diagnostic history management and filtering. The proposed system highlights several strengths, including the effective use of transfer learning, preprocessing, and data augmentation to enhance classification performance, as well as its applicability in resource-constrained environments. However, certain limitations remain. Misclassification risks persist, particularly between visually similar classes such as bacterial and viral pneumonia, and the system still requires clinical oversight, as it cannot replace medical experts. Additionally, reliance on publicly available datasets may limit generalizability to real-world clinical settings. Future work will focus on improving model accuracy and robustness through advanced architectures (e.g., Vision Transformers) and ensemble learning, as well as expanding datasets with real-world clinical data to enhance diversity. Further optimization of system performance and scalability, alongside the integration of additional clinical support features, will be explored. Ensuring ethical compliance, data security, and rigorous real-world validation will be essential steps toward practical deployment and potential commercialization.

REFERENCES

- Apostolopoulos, I. D., & Mpesiana, T. A. (2020). Covid-19: Automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, 43(2), 635–640. <https://doi.org/10.1007/s13246-020-00865-4>
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- Chandra, M. A., & Bedi, S. S. (2021). Survey on SVM and their application in imageclassification. *International Journal of Information Technology*, 13(5), 1–11. <https://doi.org/10.1007/s41870-017-0080-1>
- Dalvi, O. M. (2022). *Lungs disease dataset (4 types)*. Kaggle.
- Genuer, R., & Poggi, J.-M. (2020). *Random Forests*. https://doi.org/10.1007/978-3-030-56485-8_3
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). *CVPR 2016 Open Access Repository*. https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25. <https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html>
- Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A. W. M., van Ginneken, B., & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42, 60–88. <https://doi.org/10.1016/j.media.2017.07.005>
- Mahamud, E., Fahad, N., Assaduzzaman, M., Zain, S. M., Goh, K. O. M., & Morol, Md. K. (2024). An explainable artificial intelligence model for multiple lung diseases

- classification from chest X-ray images using fine-tuned transfer learning. *Decision Analytics Journal*, 12, 100499.
<https://doi.org/10.1016/j.dajour.2024.100499>
- Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D., Bagul, A., Langlotz, C., Shpanskaya, K., Lungren, M. P., & Ng, A. Y. (2017). *CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning* (arXiv:1711.05225). arXiv.
<https://doi.org/10.48550/arXiv.1711.05225>
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). *CVPR 2018 Open Access Repository*. Retrieved April 19, 2026, from
https://openaccess.thecvf.com/content_cvpr_2018/html/Sandler_MobileNetV2_Inverted_Residuals_CVPR_2018_paper.html
- Simonyan, K., & Zisserman, A. (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition* (arXiv:1409.1556). arXiv.
<https://doi.org/10.48550/arXiv.1409.1556>
- Tan, M., & Le, Q. V. (2020). *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks* (arXiv:1905.11946). arXiv.
<https://doi.org/10.48550/arXiv.1905.11946>
- Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (2018). *ECCV 2018 Open Access Repository*.
https://openaccess.thecvf.com/content_ECCV_2018/html/Sanghyun_Woo_Convolutional_Block_Attention_ECCV_2018_paper.html
- World Health Organization. (2024). *World health statistics 2024: Monitoring health for the SDGs, Sustainable Development Goals*. World Health Organization.