Medical Image Classification for Pneumonia Detection Using Deep Learning

1st P. Soujanya, 2nd K. Vardhan, 3rd D. Hemanandh, 4th G. Hemanth

1st Assistant Professor, 2nd–4th B.Tech Students

Department of Information Technology, ACE Engineering College (Autonomous), Hyderabad, Telangana, India

Abstract — Respiratory illnesses like pneumonia continue to be a major threat to global health, particularly among children and the elderly. Timely detection directly influences patient outcomes, yet conventional diagnostic methods remain slow and heavily dependent on specialist availability. This paper introduces an end-to-end AI-powered web application that automates the analysis of chest X-ray images using a VGG16-based Convolutional Neural Network (CNN). The model is trained on a four-class dataset covering Normal, Bacterial Pneumonia, Viral Pneumonia, and COVID-19 cases, reaching approximately 86% accuracy on a held-out test set. To bridge the gap between raw model output and clinical understanding, the system incorporates the Google Gemini API, which converts prediction probabilities into straightforward clinical narratives. A React.js front end lets healthcare workers upload images and instantly view labelled results together with Grad-CAM heatmaps that visually indicate the affected lung regions. All data — user profiles, uploaded scans, and generated reports — are persisted in a MySQL database. The outcome is a fast, interpretable, and scalable diagnostic assistant that can meaningfully support early pneumonia screening, even in low-resource or rural healthcare environments.

Keywords: *deep learning, pneumonia detection, convolutional neural network, VGG16, chest X-ray, explainable AI, Grad-CAM, Google Gemini API, medical image classification.*

1. INTRODUCTION

Pneumonia is a bacterial or viral infection that inflames lung tissue and fills the alveoli with fluid, making breathing painful and difficult. It is consistently ranked among the top causes of death in children under five years old and remains a serious concern for older adults and immunocompromised individuals. The World Health Organization estimates that the disease is responsible for roughly 14% of all deaths in young children worldwide — a figure that underscores just how critical fast and reliable diagnosis is.

The current standard of care still relies heavily on a trained radiologist visually examining chest X-rays for signs such as increased opacity, consolidation, or pleural effusion. While skilled clinicians perform this task competently, the workflow is inherently time-consuming, somewhat subjective, and constrained by the availability of qualified personnel. In many rural or resource-limited settings, patients may wait hours or even days for a radiological report — delays that can prove fatal, particularly when bacterial pneumonia or COVID-19 is involved.

Convolutional Neural Networks (CNNs) have demonstrated a genuine ability to detect pathological patterns in medical imagery with speed and accuracy that rivals experienced radiologists. Building on these advances, the present work develops a complete, deployable diagnostic platform that classifies chest X-rays into four categories, explains each classification in plain language, and maintains a secure record of every analysis performed. The system is framed as a decision-support tool rather than a replacement for the clinician — its purpose is to make expert-quality screening more accessible where specialists are hard to find.

2. LITERATURE SURVEY

Research at the intersection of deep learning and medical imaging has accelerated significantly over the past decade. Rajpurkar et al. (2017) presented CheXNet, a DenseNet-121 architecture trained on over 100,000 chest radiographs. The model matched or surpassed radiologist performance on pneumonia detection but provided no explanation for its predictions, which limited clinical adoption. Kermany et al. (2018) applied transfer learning to pre-trained CNNs and achieved strong accuracy with shorter training cycles, although inheriting fixed pre-trained representations introduced inflexibility when applied to non-standard imaging conditions.

Stephen et al. (2019) designed a purpose-built CNN for pneumonia identification, demonstrating effective feature extraction from radiographic data, though the approach demanded carefully curated and extensively annotated training sets. Liang and Zheng (2020) studied the impact of augmentation strategies — rotations, flips, and scale changes — to improve generalisation on limited datasets. While augmentation helped, the authors noted that aggressive transformations on medical scans can introduce artefacts that degrade rather than enhance model behaviour.

Work between 2021 and 2024 has increasingly focused on explainability and production readiness. Guo et al. (2023) leveraged weakly supervised labels and attention mechanisms to surface the image regions driving a prediction, improving interpretability for clinicians. Singh et al. (2024) applied Vision Transformers to chest X-rays, showing competitive accuracy alongside stronger global contextual reasoning. Angara et al. (2024) further pushed performance through ensemble methods that combine CNN and transformer outputs. Despite these advances, the majority of reported systems remain research prototypes: they lack user-friendly interfaces, persistent storage, and production-ready explanation pipelines. The present work directly addresses this gap.

3. EXISTING SYSTEMS AND PROPOSED APPROACH

3.1 Limitations of Current Systems

Most automated pneumonia detection systems in the literature share a familiar set of shortcomings. They are typically standalone models without a complete user-facing workflow, meaning clinicians cannot simply feed in a scan and receive a usable result. Predictions are often presented as raw probability vectors with no contextual explanation, which makes it difficult for medical professionals to evaluate or act on the output with confidence. Few systems maintain any persistent record of predictions, complicating audit trails and longitudinal patient tracking. The gap between a well-performing research model and a tool a non-specialist can actually use remains wide.

3.2 The Proposed System

The system proposed here takes an end-to-end perspective that begins at image upload and ends at a downloadable diagnostic report. The analytical core is a VGG16-based CNN fine-tuned on a curated multi-class chest X-ray dataset covering four conditions. This model is served via a Python FastAPI backend responsible for image ingestion, normalisation, and inference. Following classification, the predicted label and associated confidence score are forwarded to the Google Gemini API, which produces a concise plain-language clinical narrative. Results — including Grad-CAM heatmaps highlighting pathologically relevant lung regions — are presented through a React web interface. A MySQL database logs every interaction, giving the platform a verifiable history of all diagnoses.

The processing pipeline flows as follows: User Upload → Image Preprocessing → CNN Classification → Gemini Explanation → Report Generation → Database Storage. Each stage is loosely coupled, allowing individual components such as the classification model or the explanation engine to be upgraded independently without rebuilding the entire system.

4. SYSTEM ARCHITECTURE

The application follows a standard three-tier design. The Presentation Layer is a React.js single-page application providing an intuitive interface for image uploads, prediction display, historical diagnosis browsing, and report downloads. The Application Layer contains the FastAPI backend along with the TensorFlow inference engine; it coordinates calls to the Gemini API and handles user authentication. The Data Layer is a normalised MySQL schema that stores user accounts, uploaded images, prediction records, AI-generated explanations, and generated reports.

4.1 Data Flow

At the top level, the system receives a chest X-ray and returns a diagnosis with an explanation. Decomposed one level further, the process separates into distinct stages — preprocessing, CNN inference, narrative generation, and database write — each of which can be monitored and tested in isolation. This decomposition also forms the basis for structured integration testing across the system boundaries.

4.2 UML Modelling

A Use Case Diagram identifies three actors (User, Gemini API, MySQL Database) and four core scenarios (Upload Image, View Prediction, Download Report, Manage History). The Class Diagram defines the key entities — User, Image, Prediction, and Report — along with their attributes and relationships. A Sequence Diagram traces the complete message exchange from the browser through the backend, CNN model, and Gemini API and back. An Activity Diagram maps the conditional logic governing authentication, inference, explanation, and storage.

5. MODEL DEVELOPMENT

5.1 Dataset

Training data were drawn from a publicly available chest X-ray repository containing images labelled across four categories: Normal, Bacterial Pneumonia, Viral Pneumonia, and COVID-19. Class imbalances were mitigated through stratified sampling and targeted data augmentation. Images were resized to 224×224 pixels to match the VGG16 input specification, then normalised by subtracting the ImageNet channel means and dividing by the corresponding standard deviations.

5.2 Architecture and Training

VGG16 pre-trained on ImageNet was selected as the base network for its well-documented representational power on visual classification tasks. The original fully connected classification head was removed and replaced with a Global Average Pooling layer, a 256-unit Dense layer with ReLU activation, a 0.5 Dropout layer for regularisation, and a four-unit Softmax output layer corresponding to the four target classes.

Training was conducted in two phases. In the first phase, the VGG16 convolutional base was frozen and only the new classification head was updated, using a learning rate of 1×10^{-4}. This allowed the head to stabilise before any modification to the pre-trained weights. In the second phase, the top eight convolutional layers were unfrozen and the entire network was fine-tuned at a reduced learning rate of 1×10^{-5} to refine lower-level feature representations without overwriting the useful features learned during ImageNet pre-training. Early stopping, learning rate reduction on plateau, and model checkpointing were applied throughout both phases.

5.3 Explainability

Two complementary mechanisms provide interpretability. Grad-CAM (Gradient-weighted Class Activation Mapping) generates a spatial heatmap that is overlaid on the original X-ray, highlighting the anatomical regions that most strongly influenced the classification. The Google Gemini API then converts the predicted class and confidence score into a concise clinical narrative — making the output interpretable for users who may not have a radiological background and prefer a natural-language summary over a probability score.

6. SYSTEM REQUIREMENTS

6.1 Software

Category	Details
Frontend	React.js, HTML5, CSS3, JavaScript
Backend	Python 3.8+, FastAPI
Machine Learning	TensorFlow / Keras (VGG16), OpenCV, NumPy, Pandas, Scikit-learn
Explainable AI	Google Gemini API
Database	MySQL
Visualization	Matplotlib, Grad-CAM
IDE / Environment	VS Code, Jupyter Notebook; Windows / Linux / macOS

6.2 Hardware

Purpose	Specification
Development	Intel i5 or higher, 8 GB RAM, 256 GB SSD
Deployment	Dual-core 2.5 GHz+, 4 GB RAM, 50 GB storage
End-user Device	Any device with a modern browser and internet access

7. RESULTS AND EVALUATION

After completing both phases of training and fine-tuning, the model reached an overall test accuracy of approximately 86% across the four classes. The two-phase training strategy — freezing the base before progressively unfreezing it — was critical in preventing the catastrophic forgetting that can occur when pre-trained weights are modified too aggressively at the outset.

7.1 Training Behaviour

Figure 1 shows the accuracy and loss curves across 15 training epochs. Validation accuracy tracked training accuracy closely throughout, indicating that the model generalised well rather than simply memorising the training set. Loss values for both splits converged smoothly, further confirming stable learning dynamics without notable overfitting.

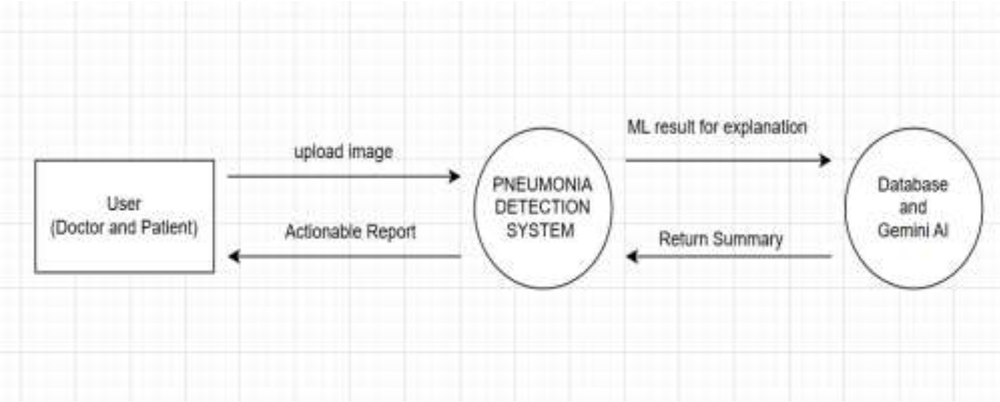

Figure 1: Training and validation accuracy (left) and loss (right) across 15 epochs.

7.2 Confusion Matrix

Figure 2 presents the confusion matrix for the test set. COVID-19 and Normal cases were classified with near-perfect accuracy (297/300 and 295/300, respectively). Bacterial Pneumonia achieved a strong result at 270/300. Viral Pneumonia proved the most challenging class, with 160/300 correct classifications; a notable proportion of viral cases were confused with bacterial pneumonia, which is consistent with the considerable visual overlap between these two conditions in chest X-ray imaging.

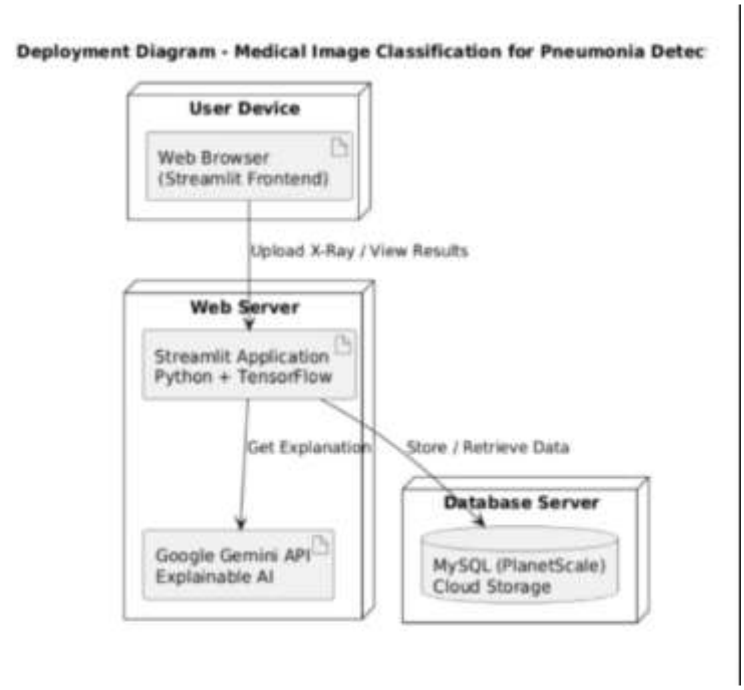

Figure 2: Confusion matrix showing per-class prediction counts on the test set.

7.3 ROC Curves

Figure 3 displays the multi-class ROC curves. Both COVID-19 and Normal categories achieved an AUC of 1.00, reflecting excellent discriminative ability. Bacterial Pneumonia attained an AUC of 0.95 and Viral Pneumonia reached 0.94 — both well above the acceptable clinical threshold. The strong AUC values across all four classes confirm that the model reliably separates each condition from the remaining categories.

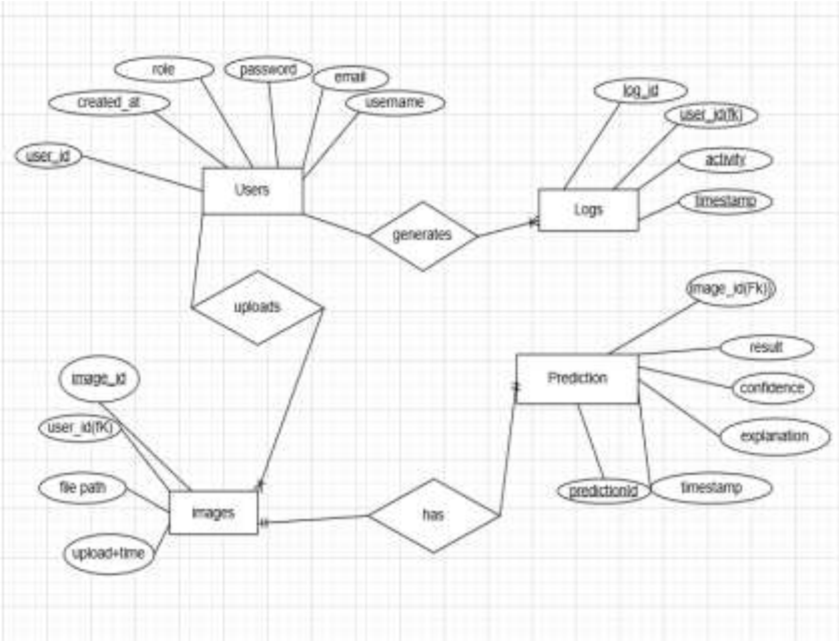

Figure 3: Multi-class ROC curves with AUC values for all four classification categories.

7.4 Unit and Integration Testing

Individual modules were validated in isolation before integration. The preprocessing pipeline correctly resized and normalised images across all test inputs. The CNN inference engine returned valid predictions and confidence scores for every accepted input format. The Gemini API integration consistently produced readable, contextually appropriate clinical narratives. Database operations — both read and write — completed reliably under normal and edge-case conditions.

End-to-end integration tests confirmed that the full data path from the React frontend through the FastAPI backend, CNN model, and database functioned correctly. Error handling was specifically exercised: when the Gemini API was intentionally slowed or disabled, the system retried and surfaced a graceful error message rather than failing silently. Under standard network conditions, a complete diagnosis cycle — from upload to displayed result — completed within approximately three seconds.

8. CONCLUSION

This work demonstrates that deep learning can be moved from a research environment into a practical, deployable clinical tool without sacrificing interpretability or usability. The VGG16-based CNN classifies chest X-rays across four pathological categories with approximately 86% accuracy, while the Gemini API gives every prediction a human-readable clinical context. The React interface and MySQL backend round out the system into a complete diagnostic assistant that healthcare workers can use without any prior knowledge of machine learning.

The project also makes a broader point: interpretability and ease of use are not optional features that can be added later — they are prerequisites for real-world adoption. A model that cannot account for its own decisions, or that demands a software engineer to operate, is unlikely to be trusted or used by clinicians regardless of how high its accuracy is. By treating transparency and accessibility as primary design requirements from the outset, this work provides a template for AI-assisted diagnostic systems that are not only accurate, but also trustworthy and practically useful in the field.

Future extensions could include detection of additional respiratory diseases such as tuberculosis and lung cancer, direct integration with hospital Picture Archiving and Communication Systems (PACS), deployment on low-power edge hardware for fieldwork, and adoption of more capable CNN–Transformer hybrid architectures for richer feature extraction.

. ACKNOWLEDGEMENT

The authors wish to thank Mrs. P. Soujanya, Assistant Professor, Department of Information Technology, ACE Engineering College, for her expert mentorship, constructive critique, and sustained encouragement throughout every stage of this project. Gratitude is also extended to Dr. S. Mani Kuchibhatla, Head of the Department of Information Technology, and to the college management for making the necessary infrastructure and resources available. Special appreciation goes to family members and peers whose unwavering support made this work possible.

.

REFERENCES

[1] Angara, S., et al. "A Novel Method to Enhance Pneumonia Detection via Model-Level Ensembling of CNN and Vision Transformer." International Journal of Computer Vision Research, 2024.

[2] Guo, Y., Zhang, H., & Liu, X. "Diagnosis and Detection of Pneumonia Using Weak-Label Deep Learning." IEEE Transactions on Medical Imaging, 2023.

[3] Kermany, D., Zhang, K., & Goldbaum, M. "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning." Cell, 2018.

[4] Liang, G., & Zheng, L. "A Transfer Learning Method with Deep Residual Network for Pediatric Pneumonia Diagnosis." Computer Methods and Programs in Biomedicine, 2020.

[5] Simard, P., Steinkraus, D., & Platt, J. "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis." Proceedings of ICDAR, 2003.

[6] Singh, R., Sharma, A., & Verma, P. "Efficient Pneumonia Detection Using Vision Transformers on Chest X-Rays." Journal of Medical Imaging and Health Informatics, 2024.

[7] Stephen, O., et al. "An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare." Journal of Healthcare Engineering, 2019.

[8] Rajpurkar, P., et al. "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning." arXiv preprint arXiv:1711.05225, 2017.