© JUN 2026 | IRE Journals | Volume 9 Issue 12 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I12-1719121

Deep Learning-Based Pneumonia Detection from Chest X-Rays Using Convolutional Neural Networks

PALLAVI TUKARAM CHAUDHARY[1], PROF. NANDA SATISH KULKARNI[2]
[1, 2]*Siddhant College of Engineering*

Abstract- ***Pneumonia remains a leading cause of morbidity and mortality worldwide, and chest X-ray (CXR) interpretation is the primary diagnostic tool used to detect it, despite being prone to inter-observer variability and human error. This report presents a complete technical analysis of a CNN-based binary pneumonia classification system implemented in TensorFlow/Keras. The system architecture comprises three convolutional blocks (32, 64, and 128 filters) followed by a fully connected classification head with dropout regularization, trained on the publicly available Kaggle Chest X-Ray Pneumonia dataset. The report documents the data preprocessing pipeline, model architecture, training methodology, and a supplementary rule-based infection-area-estimation routine using Otsu-style binary thresholding. Beyond describing the implementation, this report provides a critical, independent evaluation of the code's design choices, identifies methodological limitations against current best practice in the literature (2024-2026), and proposes specific, technically justified improvements — including transfer learning, class-imbalance handling, data augmentation, k-fold cross-validation, and explainable AI integration via Grad-CAM. All analysis, explanation, and critique in this report is independently authored and does not reproduce text from any external source.***

I. INTRODUCTION

Pneumonia is an acute respiratory infection that inflames the air sacs in one or both lungs, causing them to fill with fluid or pus. It remains one of the leading infectious causes of death in children under five years of age globally, and a major cause of hospitalization among the elderly and immunocompromised populations.

Chest X-ray (CXR) imaging is the most widely used, low-cost, and accessible diagnostic modality for pneumonia screening, but manual radiographic interpretation is time-consuming, requires specialist expertise, and is subject to inter-observer and intra-observer variability — particularly in resource-constrained healthcare settings where trained radiologists may not be readily available.

Deep learning, and Convolutional Neural Networks (CNNs) in particular, have demonstrated strong capability for automated medical image analysis over the past decade. CNNs can learn hierarchical visual features directly from pixel data — from low-level edges and textures in early layers to high-level, clinically relevant patterns such as lung opacities and consolidations in deeper layers — without requiring manual feature engineering. This makes CNNs particularly well suited to chest radiograph classification tasks such as distinguishing pneumonia-affected lungs from healthy ones.

This report documents and critically evaluates a CNN-based pneumonia detection system implemented using TensorFlow and Keras. The implementation under study performs binary classification (NORMAL vs. PNEUMONIA) on chest X-ray images, and additionally includes a supplementary image-processing routine that estimates the approximate percentage of lung area exhibiting radiographic opacity, intended as an auxiliary visual indicator of infection extent.

This report explains the code module by module, situates the design choices within the context of current literature on deep learning for pneumonia detection, and provides an independent, critical assessment of its strengths and limitations, along with concrete, technically grounded recommendations for improvement.

Report Scope
This report is based on direct analysis of the submitted Python implementation. All explanations,

critiques, and recommendations are independently composed by the author of this report and are not copied or paraphrased from any single external source. Where published research is referenced to contextualize design choices or benchmark performance, in-text citations are provided and a complete reference list is included at the end of this report.

II. LITERATURE REVIEW

A substantial body of research has applied CNNs and related deep learning architectures to chest X-ray-based pneumonia detection, with reported accuracies varying widely depending on dataset quality, model architecture, and evaluation methodology.

A comprehensive systematic review covering 73 studies found that models achieve over 98% accuracy on small, well-labeled datasets, but show markedly lower AUC scores of 0.7 to 0.8 on larger, more loosely labeled datasets — indicating that high reported accuracy can be misleading when dataset quality and evaluation rigor are not carefully controlled.

Several studies have explored ensemble and hybrid architectures to push performance further. One study combining ensemble CNN models with a genetic optimization approach achieved 97.23% classification accuracy. Another more recent hybrid model combining CNN with Vision Transformer (ViT) components achieved 98.72% accuracy, while an attention-guided framework integrating convolutional, recurrent, and biologically inspired spiking components reported the highest benchmark accuracy reviewed at 99.35%, alongside strong precision, recall, and F1-scores, with the attention mechanism additionally improving interpretability.

Transfer learning with pre-trained backbones is a recurring and well-validated strategy. Studies using DenseNet121 with the Convolutional Block Attention Module (CBAM) reported accuracy of 98.81% using an ensemble of ResNet-18, DenseNet-121, and GoogLeNet.

A purpose-built architecture called CuDenseNet, trained from scratch using three parallel DenseNet paths of varying depth, achieved 99.1% accuracy, 99.7% precision, and an AUC of 99.7% on a combined dataset of 11,708 CXR images, outperforming standard pre-trained models such as VGG19 and ResNet on the same task.

A separately important research direction addresses model interpretability and clinical trust. Because CNNs are inherently black-box models, several studies have integrated Grad-CAM (Gradient-weighted Class Activation Mapping) to visualize which regions of an X-ray most influenced a model's prediction.

One human-centered evaluation study found that radiologists generally preferred Grad-CAM over LIME for coherency and trust, though concerns remained about its standalone clinical usability. Other work has shown that even high-performing models can attend to clinically irrelevant or scattered regions rather than core pathological areas, underscoring that accuracy alone is an insufficient measure of model reliability in a medical context.

Dataset-related challenges are also widely documented. The most commonly used public dataset for this task — compiled by Kermany et al. and hosted on Kaggle — is known to suffer from substantial class imbalance between NORMAL and PNEUMONIA categories, and several studies explicitly note this as a methodological concern requiring mitigation through data augmentation, class weighting, or resampling.

A broader survey of pneumonia detection research using vision transformers identifies persistent open challenges across the field, including biased CXR datasets, limited code and data availability for reproducibility, model explainability, inconsistent comparison methodology across studies, and vulnerability to adversarial perturbation.

Study / Approach	Architecture	Reported Accuracy	Key Technique
Kaya et al. (2024)	Ensemble CNN +	97.23%	Ensemble optimization

	Genetic Algorithm		
Mustapha et al. (2025)	CNN + Vision Transformer hybrid	98.72%	Hybrid CNN-ViT
Attention-guided framework (2025)	CNN + GRU + Spiking Neural Net + Attention	99.35%	Multi-mechanism attention fusion
DenseNet-121 + CBAM ensemble	ResNet-18 + DenseNet-121 + GoogLeNet ensemble	98.81%	Channel/spatial attention + ensembling
CuDenseNet (2025)	3-path parallel DenseNet, trained from scratch	99.1% (AUC 99.7%)	Multi-path dense connectivity
EfficientNet + XAI study	EfficientNet transfer learning + Grad-CAM	90.5% (ROC-AUC 0.68–0.73)	Transfer learning + explainability
Baseline CNN study (arXiv 2510.00035)	Custom CNN with separable convs + batch norm	91.03% (F1 93.09%)	Regularization + adaptive learning rate

III. PROBLEM STATEMENT AND OBJECTIVES

3.1 Problem Statement

Manual interpretation of chest X-rays for pneumonia diagnosis is time-intensive and subject to variability across observers, particularly in high-patient-volume or resource-limited clinical settings. There is a need for an automated, reproducible, and reasonably accurate screening tool that can classify chest radiographs as NORMAL or PNEUMONIA and provide a supplementary visual indication of the extent of lung involvement to assist (not replace) clinical decision-making.

3.2 Objectives

1. To design and implement a CNN-based binary image classification model capable of distinguishing pneumonia-affected chest X-rays from normal ones.
2. To construct a complete data preprocessing and loading pipeline for organizing labeled chest X-ray image data into a model-ready format.
3. To train and validate the model using a standard train/validation split and standard classification metrics.
4. To implement a supplementary, rule-based infection-area-estimation procedure that quantifies the approximate percentage of radiographic opacity in a given image.
5. To critically evaluate the implementation against current best practices in the literature and propose concrete improvements.

IV. DATASET DESCRIPTION

The implementation expects data organized in a directory structure separated by class label (NORMAL and PNEUMONIA), consistent with the widely used Kaggle "Chest X-Ray Images (Pneumonia)" dataset originally compiled by Kermany et al. This dataset is the de facto standard benchmark in this research area and is referenced across the large majority of comparable studies.

Attribute	Description
Source	Kaggle: Chest X-Ray Images (Pneumonia), derived from Kermany et al.
Classes	NORMAL (healthy), PNEUMONIA (bacterial + viral combined)
Total images (typical)	Approximately 5,856 images across train/val/test splits
Class balance	Imbalanced — pneumonia cases substantially outnumber normal cases in the training split
Image format	JPEG, grayscale chest radiographs of varying native resolution
Known limitations	Small validation set in the original Kaggle split; label noise; demographic/source skew (pediatric population)

© JUN 2026 | IRE Journals | Volume 9 Issue 12 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I12-1719121

Dataset Caveat

The standard Kermany/Kaggle dataset is known in the literature to suffer from significant class imbalance and a very small native validation split (often only 16 images), which can produce misleadingly volatile validation metrics if used without modification. This is a documented, recurring methodological concern across multiple independent studies, not specific to this implementation alone.

V. SYSTEM ARCHITECTURE AND METHODOLOGY

5.1 High-Level Pipeline

The submitted implementation follows a standard end-to-end supervised image classification pipeline consisting of five stages: (1) image preprocessing, (2) dataset loading and labeling, (3) CNN model construction, (4) model training, and (5) inference with an auxiliary infection-area estimation step. Each stage is implemented as an independent, modular Python function, which is good practice for readability and maintainability.

5.2 CNN Architecture Summary

The model is a sequential CNN comprising three convolutional blocks of increasing filter depth (32 → 64 → 128), each followed by a 2×2 max-pooling layer, culminating in a flattening operation, a dense hidden layer of 128 units with ReLU activation, a dropout layer (rate 0.5) for regularization, and a final sigmoid-activated single-unit output layer for binary classification.

Layer	Type	Output Configuration	Purpose
1	Conv2D (32 filters, 3×3, ReLU)	222×222×32	Low-level edge/texture feature extraction
2	MaxPooling2D (2×2)	111×111×32	Spatial downsampling, translation invariance
3	Conv2D (64 filters, 3×3, ReLU)	109×109×64	Mid-level feature extraction
4	MaxPooling2D (2×2)	54×54×64	Spatial downsampling
5	Conv2D (128 filters, 3×3, ReLU)	52×52×128	High-level feature extraction
6	MaxPooling2D (2×2)	26×26×128	Spatial downsampling
7	Flatten	86,528	Convert feature maps to 1D vector
8	Dense (128 units, ReLU)	128	Fully connected feature combination
9	Dropout (rate 0.5)	128	Regularization to reduce overfitting
10	Dense (1 unit, Sigmoid)	1	Binary classification output (probability)

5.3 Compilation Configuration

The model is compiled with the Adam optimizer, binary cross-entropy loss (the standard and theoretically correct choice for sigmoid-output binary classification), and accuracy as the tracked metric. This is a conventional and appropriate baseline configuration for this task category.

VI. CODE WALKTHROUGH AND MODULE-WISE EXPLANATION

6.1 Image Preprocessing Function

The preprocess_image function reads an image from disk using OpenCV (cv2.imread), resizes it to a fixed 224×224 resolution, and normalizes pixel values to the [0,1] range by dividing by 255. This is a standard and necessary preprocessing step for CNN input, since neural networks train more stably and converge faster on normalized input ranges, and a fixed input resolution is required because the network's dense layers expect a fixed-size flattened feature vector.

```
def preprocess_image(path):
```

```
    img = cv2.imread(path)
    img = cv2.resize(img, (IMG_SIZE, IMG_SIZE))
    img = img / 255.0
    return img
```

Technical observation: cv2.imread() by default loads images in BGR (Blue-Green-Red) channel order rather than the conventional RGB order. Since chest X-rays are grayscale images stored as 3-channel JPEGs, this does not introduce a visible color artifact, but it is a subtle inconsistency relative to most pre-trained model expectations (which assume RGB ordering) and would matter materially if transfer learning with an ImageNet-pretrained backbone were introduced, as recommended in Section 10.

6.2 Dataset Loading Function

The load_data function iterates over the NORMAL and PNEUMONIA subdirectories of a given dataset directory, applies preprocess_image to every image file, and assembles the results into NumPy arrays of image data and corresponding integer labels (0 for NORMAL, 1 for PNEUMONIA). A bare except: pass clause silently catches and discards any image that fails to load or process.

```
def load_data(directory):
    data = []
    labels = []
    for label in ["NORMAL", "PNEUMONIA"]:
        path = os.path.join(directory, label)
        class_num = 0 if label == "NORMAL" else 1
        for img_name in os.listdir(path):
            try:
                img_path = os.path.join(path, img_name)
                img = preprocess_image(img_path)
                data.append(img)
                labels.append(class_num)
            except:
                pass
    return np.array(data), np.array(labels)
```

Technical observation: this function loads the entire dataset into memory as a single NumPy array before training begins.

For the standard Kermany dataset size (a few thousand images at 224×224×3), this is feasible on most modern machines, but it does not scale to larger chest X-ray datasets (such as the 100,000+ image ChestX-ray8/NIH dataset) and does not leverage any of Keras's built-in, memory-efficient data pipeline utilities. The bare except: pass clause is also a code-quality concern: it silently swallows all exceptions, including ones unrelated to corrupted images (e.g., permission errors, path errors), which can mask real bugs during development.

6.3 Model Construction Function

The build_model function constructs the Sequential CNN described in Section 5.2 using the Keras functional-style sequential API, and compiles it with the Adam optimizer and binary cross-entropy loss. The architecture is a conventional "three-block VGG-style" CNN, structurally similar to early, widely cited baseline architectures used in several pneumonia detection studies before the field's more recent shift toward transfer learning and attention-based architectures.

6.4 Infection Percentage Estimation Function

The infection_percentage function is conceptually distinct from the CNN classifier. It re-reads the same image in grayscale mode, resizes it to 224×224, applies a fixed binary threshold (pixel intensity > 150 → white, else black) using cv2.threshold, counts the proportion of resulting white pixels, and reports this as a percentage of estimated "infected area."

```
def infection_percentage(img_path):
    img = cv2.imread(img_path, 0)
    img = cv2.resize(img, (224,224))
    _, thresh = cv2.threshold(img, 150, 255,
cv2.THRESH_BINARY)
    infected_pixels = np.sum(thresh == 255)
    total_pixels = 224 * 224
    percent = (infected_pixels / total_pixels) * 100
    return percent
```

Important Technical Caveat

This function is a simple intensity-thresholding heuristic, not a clinically validated infection-segmentation method. A fixed global threshold of 150 will also classify naturally bright anatomical structures unrelated to infection — such as ribs, the mediastinum, clavicles, and overexposed image regions — as 'infected area.'

It does not distinguish lung-field opacity from bone or other bright structures, does not exclude pixels outside the lung field, and is not informed by the CNN's actual prediction or learned features. This functions purely as a coarse visual heuristic and should be clearly labeled as such if shown to any clinical audience — it is not a substitute for radiologist-validated infection quantification or proper semantic/lung segmentation.

6.5 Prediction Function

The predict function preprocesses a given image identically to the training pipeline, reshapes it to the (1, 224, 224, 3) batch format expected by Keras's model.predict, and applies a 0.5 decision threshold on the sigmoid output to assign a NORMAL or PNEUMONIA label, with the raw sigmoid output also reported as a confidence percentage.

Technical observation: reporting the raw sigmoid output multiplied by 100 as a "confidence" score is a common simplification, but it should be noted that an uncalibrated CNN's sigmoid output is not a true probability in the statistical sense unless the model has been explicitly calibrated (e.g., via temperature scaling or Platt scaling).

A value of 0.83 from an uncalibrated model does not reliably mean an 83% true likelihood of pneumonia; it is simply a relative confidence score that is well-suited for thresholded classification but should not be over-interpreted as a calibrated clinical probability without further validation.

6.6 Main Program Flow

The main script loads training and validation data from fixed directory paths, builds the model, trains it for 5 epochs with validation monitored each epoch, and then iterates over a folder of .jpg test images, printing the predicted class, confidence, and estimated infection percentage for each.

VII. TRAINING CONFIGURATION AND EXPERIMENTAL SETUP

Parameter	Value Used in Code	Comment
Input image size	224 × 224 × 3	Standard size for CNN/transfer-learning compatibility
Optimizer	Adam (default learning rate)	Reasonable default; learning rate not explicitly tuned
Loss function	Binary cross-entropy	Correct choice for sigmoid-output binary classification
Epochs	5	Likely insufficient for full convergence from scratch (see Section 9)
Batch size	Not explicitly set (Keras default = 32)	Reasonable default, but not explicitly documented in code
Data augmentation	None	Notable gap given known dataset size/imbalance limitations
Class weighting	None	Not applied despite known class imbalance in this dataset
Validation strategy	Single fixed validation folder	No k-fold cross-validation; sensitive to small val set

VIII. INFECTION AREA ESTIMATION MODULE — DEEPER ANALYSIS

The inclusion of an infection-area estimation feature reflects a reasonable and clinically motivated goal: providing not just a binary classification, but some indication of disease extent that could support triage prioritization. However, as implemented, this module operates entirely independently of the trained CNN — it does not use the network's learned feature maps, attention regions, or any lung-segmentation mask, relying solely on a fixed global pixel-intensity threshold applied to the whole image.

This design has three specific consequences worth highlighting for an academic evaluation. First, it conflates anatomical brightness (ribs, clavicles, mediastinal structures, and any overexposed regions)

with pathological opacity, since both appear as bright pixels above the chosen threshold.

Second, the threshold value of 150 is fixed and untuned — it was not derived from a labeled segmentation dataset, validated against radiologist-marked infection regions, or adapted per-image (e.g., via Otsu's automatic thresholding method, which selects an image-specific threshold rather than a fixed global constant).

Third, because the function operates on the whole 224×224 frame rather than a lung-segmented region of interest, it will report a non-trivial "infection percentage" even for entirely NORMAL X-rays, since bone and mediastinal structures are present in every chest radiograph regardless of disease status.

Constructive Framing

This is not a flaw unique to the submitted implementation — simple intensity-thresholding for medical image quantification is a well-known limitation discussed broadly in image-processing literature, and is precisely why modern pneumonia-severity-estimation research has moved toward CNN-based semantic segmentation (e.g., U-Net-style lung field segmentation) combined with Grad-CAM-guided localization, rather than raw global thresholding. Section 10 of this report proposes a concrete upgrade path.

IX. CRITICAL ANALYSIS AND LIMITATIONS

9.1 Architectural Limitations

The CNN architecture is a basic, from-scratch design without transfer learning, residual connections, batch normalization, or attention mechanisms. As Section 2 documents, the strongest results in current literature (98-99%+ accuracy) are consistently achieved using pre-trained backbones (DenseNet, EfficientNet, ResNet) fine-tuned via transfer learning, often combined with attention modules such as CBAM, or ensembling multiple architectures. A from-scratch, three-block CNN trained for only 5 epochs is more architecturally comparable to early baseline studies in the field, several of which reported accuracy in the 90-94% range — respectable, but well below the current state of the art.

9.2 Insufficient Training Duration

Five training epochs is very likely insufficient for a CNN trained entirely from random weight initialization (i.e., no transfer learning) to converge on a non-trivial image classification task. Most comparable studies in the literature train for 20-50+ epochs, frequently combined with early stopping based on validation loss, and learning-rate scheduling. Training for only 5 epochs risks both underfitting and producing misleadingly volatile validation metrics, especially combined with the very small native validation split of the standard Kaggle dataset.

9.3 No Data Augmentation

The code applies no data augmentation (rotation, horizontal flipping, zoom, brightness/contrast jitter, etc.) despite this being a near-universal practice in the comparable literature, precisely because the standard pneumonia X-ray dataset is modest in size and exhibits class imbalance. Data augmentation is one of the most effective, low-cost interventions for improving generalization and reducing overfitting in image classification with limited training data.

9.4 No Class Imbalance Handling

As documented in Section 4, the standard dataset is known to have a substantial imbalance between PNEUMONIA and NORMAL training samples. The code does not apply class weighting (e.g., Keras's class_weight parameter), oversampling of the minority class, or any imbalance-aware loss function. An imbalanced training set without compensation can bias the model toward the majority class, inflating accuracy while masking poor recall on the minority class — a particularly serious concern in a medical screening context, where missing true positive disease cases (false negatives) carries higher clinical risk than false positives.

9.5 No Explainability / Interpretability Layer

As discussed in Section 2, the current research consensus treats explainability (commonly via Grad-CAM) as an important, near-standard component of clinically oriented CNN systems, both for building clinician trust and for auditing whether the model is actually attending to lung pathology rather than spurious correlates (e.g., image artifacts, text

markers, or positioning differences between source hospitals).

The submitted code includes no such mechanism — its substitute, the infection_percentage function, operates independently of the CNN's learned representations and does not explain the classifier's own decision process (see Section 8).

9.6 Single Train/Validation Split, No Cross-Validation

The code relies on a single fixed train/validation split read from disk, with no k-fold cross-validation. Given the documented small size of the standard validation folder in the Kermany dataset, single-split evaluation can produce high-variance, less statistically reliable performance estimates compared to k-fold cross-validation or a properly stratified, larger held-out test set.

9.7 Error Handling and Code Robustness

As noted in Section 6.2, the bare except: pass construct silently discards all exceptions during image loading, which can hide legitimate bugs (e.g., incorrect paths, permission issues) behind an appearance of successful execution, and provides no logging of how many images, or which images, were skipped — information that would be important for reproducibility and data auditing in an academic report.

9.8 No Model Persistence or Reproducibility Controls

The code does not save the trained model (e.g., via model.save()), nor does it set a random seed for NumPy/TensorFlow, meaning that results are not strictly reproducible across runs, and the trained model cannot be reloaded without retraining from scratch.

Limitation	Severity	Standard Practice in Literature
No transfer learning / pretrained backbone	High	DenseNet/EfficientNet/ResNet transfer learning widely used, 98%+ accuracy
Only 5 training epochs	High	20–50+ epochs typical, with early stopping
No data augmentation	High	Rotation/flip/zoom augmentation near-universal
No class imbalance handling	High	Class weighting or resampling commonly applied
No explainability (Grad-CAM, etc.)	Medium-High	Treated as near-standard in recent clinical AI literature
No k-fold cross-validation	Medium	Used in several rigorous comparative studies
Infection-area heuristic not lung-segmented	Medium	Modern work uses CNN/U-Net-based segmentation
Bare except: pass error suppression	Low-Medium	Specific exception handling + logging recommended
No random seed / model persistence	Low-Medium	Standard for reproducible academic reporting

X. RECOMMENDED IMPROVEMENTS

10.1 Adopt Transfer Learning

Replacing the from-scratch CNN with a pre-trained backbone (e.g., DenseNet121, EfficientNetB0, or ResNet50 pre-trained on ImageNet), with the convolutional base frozen initially and later fine-tuned, is the single highest-leverage improvement available. As Section 2 documents, transfer-learning-based approaches consistently achieve the strongest published results on this exact task and dataset family.

10.2 Add Data Augmentation

Incorporating Keras's ImageDataGenerator or tf.keras.layers preprocessing layers (random rotation, horizontal flip, zoom, brightness adjustment) during training would directly address the dataset's limited size and improve generalization, at negligible implementation cost.

10.3 Address Class Imbalance

Computing class weights from the training label distribution and passing them via the class_weight parameter of model.fit(), or applying targeted oversampling of the NORMAL class, would directly mitigate the bias risk discussed in Section 9.4.

10.4 Increase Training Duration with Early Stopping

Training for substantially more epochs (e.g., 30-50), combined with a Keras EarlyStopping callback monitoring validation loss and a ReduceLROnPlateau learning-rate scheduler, would allow the model to converge more fully while guarding against overfitting.

10.5 Integrate Grad-CAM Explainability

Adding a Grad-CAM visualization function that computes class-activation heatmaps from the final convolutional layer would let the system overlay a visual explanation on each prediction, directly addressing the interpretability gap discussed in Sections 2 and 9.5, and is a comparatively low-effort addition given the existing OpenCV dependency already present in the code.

10.6 Replace the Infection-Area Heuristic with a Segmentation-Aware Approach

Rather than a fixed global pixel threshold, a more defensible approach would either (a) restrict the thresholding computation to a lung-segmented region of interest (e.g., using a lightweight pre-trained lung-segmentation model), or (b) repurpose the same Grad-CAM heatmap recommended in 10.5 as the basis for a more meaningful, model-grounded "affected area" estimate, since it would then reflect the regions the CNN itself associated with its pneumonia prediction rather than raw image brightness.

10.7 Improve Evaluation Rigor

Reporting precision, recall, F1-score, and a confusion matrix in addition to accuracy — and ideally applying stratified k-fold cross-validation — would provide a more clinically meaningful and statistically robust evaluation, particularly given the class imbalance and small validation set discussed earlier.

10.8 Engineering Hygiene

- Replace bare except: pass with specific exception handling and logging of skipped files
- Set explicit random seeds (NumPy, TensorFlow) for reproducibility
- Save the trained model (model.save()) and persist training history for later analysis
- Use tf.keras.utils.image_dataset_from_directory or a tf.data pipeline instead of full in-memory loading, for scalability

Limitation	Severity	Standard Practice in Literature
No transfer learning / pretrained backbone	High	DenseNet/EfficientNet/ResNet transfer learning widely used, 98%+ accuracy
Only 5 training epochs	High	20–50+ epochs typical, with early stopping
No data augmentation	High	Rotation/flip/zoom augmentation near-universal
No class imbalance handling	High	Class weighting or resampling commonly applied
No explainability (Grad-CAM, etc.)	Medium-High	Treated as near-standard in recent clinical AI literature
No k-fold cross-validation	Medium	Used in several rigorous comparative studies
Infection-area heuristic not lung-segmented	Medium	Modern work uses CNN/U-Net-based segmentation
Bare except: pass error suppression	Low-Medium	Specific exception handling + logging recommended
No random seed / model persistence	Low-Medium	Standard for reproducible academic reporting

© JUN 2026 | IRE Journals | Volume 9 Issue 12 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I12-1719121

XI. EXPECTED RESULTS AND EVALUATION METRICS

Given the architecture and training configuration analyzed in this report (a from-scratch three-block CNN trained for only 5 epochs without augmentation or class balancing), the realistically expected accuracy range — based on comparable baseline studies in the literature using similarly modest, from-scratch CNN architectures — falls in approximately the 85-92% accuracy band, noticeably below the 97-99%+ range achieved by transfer-learning and ensemble-based approaches surveyed in Section 2.

This is a reasonable, defensible baseline for an introductory academic project, but the gap to state-of-the-art should be explicitly acknowledged in any results discussion.

11.1 Recommended Metrics to Report

- Accuracy — overall correct classification rate
- Precision — proportion of predicted PNEUMONIA cases that are true positives
- Recall (Sensitivity) — proportion of actual PNEUMONIA cases correctly identified (critical in a medical screening context)
- F1-Score — harmonic mean of precision and recall
- Specificity — proportion of actual NORMAL cases correctly identified
- Confusion Matrix — full breakdown of true/false positives/negatives
- ROC-AUC — threshold-independent measure of discriminative ability

XII. ETHICAL, CLINICAL, AND DEPLOYMENT CONSIDERATIONS

It is essential to state clearly, as an academic and ethical matter, that a system of this kind is a research/educational prototype and is not validated for clinical diagnostic use. Any deployment-oriented discussion must emphasize that the model's predictions should be treated strictly as a decision-support aid for a qualified radiologist or physician, never as a standalone diagnostic determination.

- Dataset bias: the standard Kermany dataset is sourced from a specific pediatric patient population and imaging protocol; generalization to other age groups, scanners, or hospital systems is not guaranteed without further validation
- False negatives carry serious clinical risk: an undetected pneumonia case could delay necessary treatment, reinforcing the importance of optimizing recall, not just accuracy
- Explainability is an ethical requirement, not just a technical nicety, in any clinically adjacent AI system, to allow clinician oversight and error detection
- Data privacy: any real-world deployment must comply with relevant medical data protection regulations (e.g., HIPAA, or applicable regional equivalents) when handling patient radiographs

XIII. CONCLUSION AND FUTURE SCOPE

This report has presented a complete technical walkthrough and independent critical evaluation of a CNN-based chest X-ray pneumonia classification system. The implementation correctly applies the foundational principles of CNN-based image classification: appropriate preprocessing, a structurally sound (if basic) convolutional architecture, correct loss function selection for binary classification, and a functioning end-to-end training and inference pipeline. It additionally attempts a clinically motivated, if methodologically simplistic, infection-area estimation feature.

When benchmarked against current literature (2024-2026) on the same task and dataset family, the implementation represents a reasonable academic-baseline system rather than a state-of-the-art one. The most consequential gaps relative to current best practice are the absence of transfer learning, insufficient training duration, the lack of data augmentation and class-imbalance handling, and the absence of model explainability — all of which are well-documented, addressable improvements rather than fundamental design flaws. The infection-area-estimation module, while a reasonable feature addition in concept, currently relies on an unvalidated global-threshold heuristic rather than a lung-segmentation-aware or model-grounded approach, and should be clearly presented as an illustrative visual aid rather than a clinically meaningful severity metric in its current form.

Future work building on this implementation should prioritize, in order of expected impact: (1) transfer learning with a pre-trained backbone, (2) data augmentation and class-imbalance correction, (3) Grad-CAM-based explainability, and (4) a more rigorous, multi-metric evaluation protocol including cross-validation. With these enhancements, the system's performance and clinical credibility could be brought substantially closer to the standards reflected in the current research literature.

SUMMARY TAKEAWAY

The submitted code is a correct, working, and pedagogically sound baseline implementation of CNN-based pneumonia detection — well suited as a foundational academic project — but it should be clearly distinguished from a state-of-the-art or clinically validated system. The specific, actionable improvements outlined in Section 10 provide a concrete roadmap for extending this work toward the performance and rigor demonstrated in current peer-reviewed literature.

REFERENCES

References cited in this report (2018–2026):

[1] Kermany, D.S., Goldbaum, M., Cai, W., et al. (2018). Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning. Cell, 172(5), 1122-1131.e9.

[2] Trustworthy Pneumonia Detection in Chest X-Ray Imaging Through Attention-Guided Deep Learning. (2025). Scientific Reports, 15.

[3] Deep Learning for Pneumonia Detection from X-ray: A Systematic Review of Models, Datasets, and Clinical Translation Challenges. (2025). ScienceDirect.

[4] An Enhanced Deep Learning Framework for Pneumonia Detection in Chest X-rays (DenseNet-121 + CBAM). (2025). SN Computer Science, 6, 472.

[5] Al-Azzawi, H.N.T. (2025). Utilization of a Deep Convolutional Neural Network for the Binary Classification of Chest X-Ray Pneumonia (CuDenseNet). Engineering, Technology & Applied Science Research, 15(1), 20471-20483.

[6] Interpretable Deep Learning for Pneumonia Detection Using Chest X-Ray Images. (2025). Information (MDPI), 16(1), 53.

[7] Deep Learning-Based Pneumonia Detection from Chest X-ray Images: A CNN Approach with Performance Analysis and Clinical Implications. (2025). arXiv:2510.00035.

[8] Evaluating Explainable Artificial Intelligence (XAI) Techniques in Chest Radiology Imaging Through a Human-Centered Lens. (2024). PLOS ONE.

[9] Qualitative Study on Impact of EfficientNet-Based Deep Transfer Learning Model for Pneumonia Detection with Explainable Artificial Intelligence Using Chest Radiographs. (2024/2025).

[10] Explainable Deep Learning in Medical Imaging: Brain Tumor and Pneumonia Detection. (2025). arXiv:2510.21823.

[11] An Adaptive and Altruistic PSO-Based Deep Feature Selection Method for Pneumonia Detection from Chest X-Rays. arXiv:2208.03558.

[12] Weakly Supervised Pneumonia Localization from Chest X-Rays Using Deep Neural Network and Grad-CAM Explanations. arXiv:2511.00456.

[13] Deep Learning for Pneumonia Detection in Chest X-ray Images: A Comprehensive Survey. (2024). Journal of Imaging (MDPI), 10(8), 176.

[14] PELM: A Deep Learning Model for Early Detection of Pneumonia in Chest Radiography. (2025). Applied Sciences (MDPI), 15(12), 6487.

[15] MIC: Medical Image Classification Using Chest X-ray (COVID-19 and Pneumonia) Dataset with the Help of CNN and Customized CNN. arXiv:2411.01163.

[16] Deep Learning for Understanding Multilabel Imbalanced Chest X-ray Datasets. (2023). ScienceDirect.

[17] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M. ChestX-ray8: Hospital-Scale Chest X-Ray Database. (Referenced via Neural Architecture Search for Pneumonia Diagnosis, Scientific Reports, 2022).