https://doi.org/10.3311/PPee.43418


**Periodica Polytechnica Electrical Engineering and Computer Science**, 70(1), pp. 78–88, 2026

# Systematic Hyperparameter Optimization of Convolutional Neural Networks for Pneumonia Detection from Chest X-rays

Anthony Nwohiri[1*], David Olukolatimi[2], Ayorinde Alase[3]

[1] Department of Computer Science, Faculty of Computing and Informatics, University of Lagos, University Road, Akoka, Yaba, 101017 Lagos, Nigeria
[2] AXA Mansard, Santa Clara Court, Plot 1412 Ahmadu Bello Way, Victoria Island, 106104 Lagos, Nigeria
[3] School of Engineering and Engineering Technology, University of Arkansas at Little Rock, 2801 S. University Ave., 72204 Little Rock, AR, USA
[*] Corresponding author, e-mail: anwohiri@unilag.edu.ng

Received: 24 January 2026, Accepted: 10 April 2026, Published online: 15 April 2026

**Abstract**

Pneumonia remains a major global health challenge. It is responsible for millions of deaths annually, making it one of the leading causes of mortality worldwide, particularly in low- and middle-income countries, where access to early and accurate diagnosis is often limited. Convolutional neural networks (CNNs) have shown strong potential in medical image classification. However, the performance of these deep learning models depends primarily on the choice of hyperparameters, which determine how effectively the models learn. This study investigates the interaction between multi-factor hyperparameters, which is a major gap in large-scale optimization for medical imaging. The goal is to systematically evaluate these interactions in order to develop a CNN diagnostic model with improved pneumonia detection accuracy. Using the RSNA Pneumonia Detection Challenge dataset, four pre-trained CNN architectures (VGG16, ResNet50, InceptionV3, and MobileNetV2) were trained and tested. Our methodology progressed from single-factor ablation to multi-factor interaction analysis. Two optimization approaches, Random Search (RS) and Bayesian Optimization (BO), were applied across five key hyperparameters: learning rate, epoch count, batch size, dropout rate, and optimizer. Results revealed that learning rate and optimizer had the greatest impact on classification accuracy. InceptionV3 was found to be the best-performing model (learning rate 0.001, batch size 64, dropout 0.1, 40 epochs, Adam optimizer), with a test F1-score of 92.5% and recall of 91.8%. BO consistently outperformed RS in improving CNN performance for pneumonia detection. Overall, these findings highlight that systematic, dataset-specific hyperparameter tuning can greatly enhance the reliability of CNN-based pneumonia detection from chest X-rays.

**Keywords**

convolutional neural network, hyperparameter optimization, pneumonia detection, chest X-ray, medical imaging, deep learning

## 1 Introduction

Pneumonia is a severe respiratory infection that affects many people worldwide and it is associated with so many health issues, as well as deaths globally [1]. In low- and middle-income countries, where there is limited access to medical facilities and skilled radiologists, pneumonia is a leading cause of death in children under the age of 5 and adults over 65 [2]. When the lungs' air sac becomes inflamed and filled with fluid, pneumonia occurs, making it difficult for the victim to breathe; this could become fatal if timely treatment is not received [3].

Despite major advancements in medical imaging, numerous healthcare systems still rely on manual assessment by radiologists. Such manual interventions are bound to suffer from delays and fatigue, and to a greater extent, cause a lack of uniformity in interpretation [4]. Under such conditions, early and particularly accurate diagnosis of pneumonia becomes a very challenging task. Therefore, there is a greater need to create a computerized diagnostic tool to assist radiologists and clinicians to more efficiently and accurately detect pneumonia from radiographs.

Accurate and early detection of pneumonia is critical and offers significant clinical benefits. This is necessary for a favorable outcome after treatment [5, 6]. The first-line imaging technique routinely used in pneumonia diagnosis is chest X-rays. It is one of the most preferred diagnostic radiology tools across various clinical practices and

**Cite this article as:** Nwohiri, A., Olukolatimi, D., Alase, A. "Systematic Hyperparameter Optimization of Convolutional Neural Networks for Pneumonia Detection from Chest X-rays", Periodica Polytechnica Electrical Engineering and Computer Science, 70(1), pp. 78–88, 2026. https://doi.org/10.3311/PPee.43418

hospitals and healthcare institutions, owing to its low cost and accessibility [7]. However, even though the performance and quality of the X-rays are excellent, interpretation of the X-ray results is subjective, and opinion differences are bound to occur among different radiologists. This has driven growing interest in the application of artificial intelligence (AI), most especially deep learning, in medical imaging. Deep learning (DL) enables the automatic learning of the complex features of images that might be difficult for the human eye to catch, improving the accuracy and the uniformity of diagnosis across healthcare facilities.

While there are numerous DL approaches, convolutional neural network (CNN) has proven to be highly effective in analyzing X-rays and CT scans to detect diseases [8–10]. CNNs can automatically learn edges, textures, and patterns from images, which might be indications of the presence of diseases. The performance of CNNs models is exceptional, however, their efficiency, accuracy, convergence speed, and generalization are determined mostly by hyperparameters, such as learning rate, batch size, number of epochs, dropout ratio, and optimizer [11].

The tuning of these models is complicated by the non-linear interrelationships between these variables. For example, optimal learning rate strongly depends on the selected optimization method and batch size, such that independent optimization of these components does not reflect the multidimensional dynamics needed to achieve maximum diagnostic performance.

Although CNNs are increasingly used for disease detection in medical imaging, the influence of hyperparameter tuning on their performance has not been thoroughly explored. Lacerda et al. [12] carried out a study on hyperparameter optimization of CNNs for the diagnosis of COVID-19 pneumonia using chest CT images. However, they considered only a few parameters and a limited dataset. Gea et al. [13] explored the influence of hyperparameter tuning in the classification of MRI brain tumors, considering five configurations and leaving out how parameters interact with each other. Shahira and Negara [14] applied Bayesian Optimization (BO) to a DenseNet-169 model for three-class lung X-ray classification (normal, pneumonia, COVID-19); however, they explored only a narrow range of hyperparameters and architectures.

These studies highlight the role of hyperparameter tuning. However, there is a notable lack of large-scale, multi-model, and multi-technique investigation on the effects of hyperparameter interactions on model performance, particularly in pneumonia detection.

To address these gaps, this study makes three major contributions by systematically mapping hyperparameter interactions. First, it measures the non-linear interactions among the five key hyperparameters, directly addressing the prevalent reliance on single-factor analyses as reported in the literature. Second, it investigates the architecture-dependency of these interactions using four popular CNNs. This revealed that optimal configurations are highly model-specific instead of universal. Third, it assesses the tuning efficiency of two optimization techniques on large-dimensional medical imaging data, suggesting that BO is better suited than the traditional Random Search (RS) in traversing high dimensional parameter spaces.

## 2 Related works

Recent research shows that CNNs are becoming increasingly relevant in the detection of pneumonia based on chest X-ray images. Rajpurkar et al. [15] proposed the CheXNet model by training a 121-layer DenseNet and achieving a radiologist-level pneumonia detection accuracy. Ayan and Ünver [16] implemented transfer learning with VGG16 and VGG19 networks, demonstrating effective detection of pneumonia using a fine-tuned CNN with very minimal computing resources. It is known that convolutional networks outperform traditional computation methods in medical imaging. For instance, Pechnikov et al. [17] showed that pneumonia detection *via* CNN is both faster and more accurate than *via* the Kolmogorov complexity method.

The importance of using transfer learning in medical imaging classifiers lies with the fact that it uses pre-trained models. Kermany et al. [18] showed that transfer learning increased pneumonia detection accuracy. Brima et al. [19] reported a test accuracy of 94.0% using pre-trained weights. Chouhan et al. [20] were able to avoid overfitting on small medical datasets by using an ensemble of selected pre-trained architectures, including DenseNet121 and InceptionV3. All these studies support the development of CNN-based pneumonia detection models and show that transfer learning simplifies the implementation process.

A primary limitation of existing research is the tendency to optimize parameters in isolation. This approach neglects hyperparameter interactions (functional dependency), where a change in one parameter, e.g., batch size, shifts the optimal range for another, e.g., learning rate.

In medicine, the selection of model architectures and use of hyperparameter tuning have received increasing attention. Lacerda et al. [12] examined the hyperparameters of CNNs and the detection of COVID-19 pneumonia based on CT scans. Shahira and Negara [14] took a more organized probabilistic approach by implementing BO on a DenseNet-169 X-ray model. They reported improved classification accuracy. Gea et al. [13] investigated hyperparameter tuning on CNNs for the classification of brain tumors from MRI scans. Despite these contributions, their methods were limited in scope, mostly focusing on a single model or a narrow range of parameters. There was little attention on cross-model comparisons and the interaction of parameters.

In the context of hyperparameter tuning, there are multiple studies suggesting its importance yet there are still gaps and limitations to be resolved within these studies. Several suggested pneumonia detection models are based on fixed default hyperparameter settings rather than performing hyperparameter optimization [21–23]. These models are tested on small or skewed data distributions [24, 25], which limits their generalization in the real-world. Moreover, only few studies have compared probabilistic and randomized optimization across multiple CNN architectures to assess the particular impact of hyperparameter interactions on diagnostic performance [26–31].

To fill these gaps, five hyperparameters were assessed across four pre-trained CNN architectures (VGG16, ResNet50, InceptionV3, and MobileNetV2) using chest X-ray images. By comparing RS and BO, this research specifically isolates multi-factor hyperparameter interactions that prior literature often overlooks. This yields a structured framework for improving diagnostic accuracy without incurring unnecessary computational costs.

## 3 Methodology

### 3.1 Dataset

In the original database, there are almost 30,000 radiographs that are categorized in three groups. Since this research is specifically binary classification, the images that were labeled as "No Lung Opacity" or "Not Normal" (11,821 in total) were excluded. The rest of the data had a moderate balance of 8,851 normal cases and 6,012 lung opacity cases. A random undersampling approach was adopted to obtain a 1:1 baseline and maintain a computationally viable structure to execute over 1,500 hyperparameter optimization trials to obtain an optimal result. A fixed computational seed was used to select exactly 3,000 images randomly in each of the two target classes. This method produced the 6,000-image working dataset as presented in Table 1.

To achieve dimension consistency, all the images were resized to 224 × 224 pixels. Pixel intensities were normalized to the range 0–1. Fig. 1 shows exemplary images from the dataset for both pneumonia and normal cases.

**Table 1** Dataset distribution

| Class | Sample count |
|---|---|
| Lung opacity | 3,000 |
| Normal | 3,000 |
| Total | 6,000 |

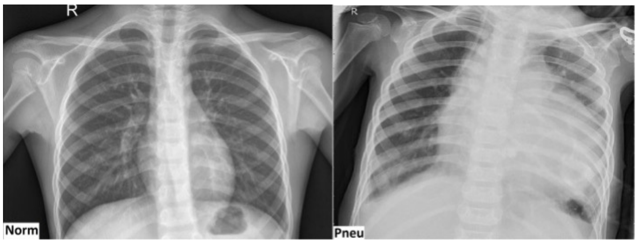


**Fig. 1** Sample chest X-ray images from the RSNA Pneumonia Detection Challenge dataset

The dataset was split into three groups: training (70%), validation (15%), and testing (15%). This division was carried out image-wise through a stratified random splitting scheme that is based on the target labels. The 1:1 class proportion of the total dataset was maintained in each subset through stratification. The partition was performed at the image-level, rather than being explicitly partitioned on a patient-by-patient basis.

Fig. 2 shows the age distribution of patients in the dataset. The distribution is heavily skewed toward adults. This is significant when comparing the results with pediatric studies. As summarized in Table 1, the dataset contains an equal number of lung opacity and normal cases, with a total of 6,000 patients.

### 3.2 CNN architecture

As already mentioned above, this study was based on four widely used and high-performing pre-trained CNN architectures (VGG16, ResNet50, InceptionV3, and MobileNetV2). They were chosen to provide a comparative baseline of data with varying network depths and number of parameters, which could be used to compare the impact of architectural variations on optimal hyperparameter settings. For each of

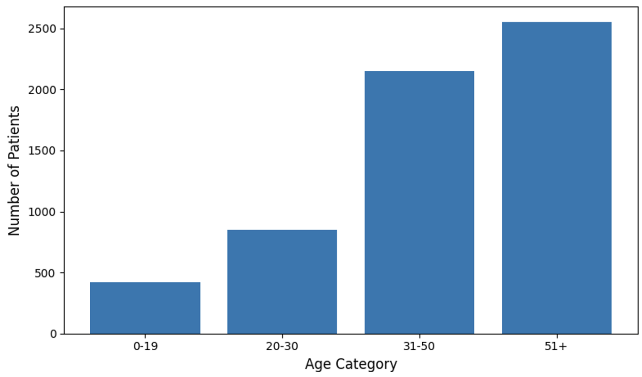


**Fig. 2** Patient age distribution

the model networks, the ImageNet dataset was used to pretrain the weights, after which the networks were customized to the pneumonia detection problem.

In order to ensure the reproducibility of the experiment, common seed (SEED = 23) was set throughout Python [32], NumPy [33], and TensorFlow [34]. Preprocessing of data involved downsizing of DICOM images to 224 × 224 pixels, color conversion to RGB, and scaling of pixels to a [0, 1] range. Training was done on a stratified split of data (70% training, 15% validation, 15% testing) of frozen base ImageNet weights and binary classification heads. All experiments such as using custom F1-score metrics, Optuna-based tuning were run on a Kaggle cloud platform with a P100 GPU whose memory growth was explicitly turned on.

### 3.3 Hyperparameters

The hyperparameters analyzed were the learning rate ranging from $1 \times 10^{-5}$ and $1 \times 10^{-1}$ batch size, 16, 32, 64, 128, and 256; optimizer type, between Adam and stochastic gradient descent (SGD); training epochs, set between 10 and 40; and dropout rate, between 0.1 and 0.5.

These intervals were chosen in order to find out the training stability thresholds. The learning rates (ranging between $1 \times 10^{-5}$ and $1 \times 10^{-1}$) and dropout rates (between 0.1 and 0.5) helped to make sure that the algorithms would explore the limits between extreme overfitting and ideal generalization of the models.

### 3.4 Optimization techniques

RS and BO were the algorithms used for optimizing the hyperparameters. RS uses several random combinations of hyperparameters and provides a diverse and broad array of potential configurations. In contrast, BO focuses on specific configurations in a more calculated manner. That is, given prior combinations and outcomes, it selects new combinations and iteratively improves on the configuration.

BO is a scientifically grounded process that can approximate an objective function to maximize F1-score on validation across the hyperparameter space. The Tree-structured Parzen Estimator (TPE) [a BO algorithm] does not use exhaustive evaluation, it builds a probabilistic surrogate model. The algorithm is based on dynamically choosing the next configuration, which will likely improve the current best F1-score by optimizing the computational efficiency.

Optimization of the four CNN classifiers was performed individually followed by extensive calculation of relevant metrics of each classifier such as accuracy and F1-score to determine the best model and optimal hyperparameter combinations for pneumonia diagnosis from chest X-ray images.

Deep CNNs are computationally impractical because exhaustive methods such as grid search can grow exponentially when applying grid search. RS and BO were embraced to cope with the cost of computation without compromising the performance of the models. The algorithms are also very effective in sampling the parameter space and discovering high-performing configurations much faster than brute-force algorithms. They are therefore well suited for compute-intensive image classification problems.

### 3.5 Experimental framework

The experimental design started by testing parameters on their own to identify baseline behaviors. Nevertheless, optimizing hyperparameters individually is insufficient because hyperparameters are highly interdependent; the ideal learning rate can change as the batch size varies. In order to take this into consideration, the analysis advanced to multi-factor interaction analysis. Although simpler and two-factor tests were done under grid search, it became necessary to switch to RS and BO to evaluate three or more interacting parameters, ensuring that computational complexity is not prohibitive and the accuracy of the combined effects is intact.

Table 2 summarizes experimental approaches, experiment counts, and total training times. The mean training time across all methods was on average 0.40 h per experiment.

## 4 Results

### 4.1 Single-factor analysis

Using one-factor-at-a-time analysis, the effects of separate hyperparameters on the F1 score were evaluated first. A statistically significant positive correlation between learning rate and F1 score was observed for VGG16 and

**Table 2** Summary of hyperparameter optimization experiments and training time

| Methodology | Algorithm | Number of optimized parameters | Number of experiments | Total model training time (h) |
|---|---|---|---|---|
| 1-factor | Grid search | 1 | 88 | 28.1 |
| 2-factor | Grid search | 2 | 478 | 167.3 |
| 3-factor | RS | 3 | 240 | 99.6 |
| | BO | 3 | 240 | 99.1 |
| 4-factor | RS | 4 | 160 | 69.6 |
| | BO | 4 | 160 | 70.4 |
| 5-factor | RS | 5 | 80 | 40.2 |
| | BO | 5 | 80 | 37.8 |
| Total | | | 1,526 | 612.1 |

ResNet50, as shown in Table 3. This indicates the importance of learning rate in model generalization. In contrast, batch size was found to have a statistically significant negative effect on the F1 score for VGG16 and InceptionV3.

#### 4.1.1 Distribution of optimized F1 scores per hyperparameter

The widest ranges of F1 score values were observed for the learning rate and number of epochs (see Fig. 3). This indicates that these hyperparameters play the most crucial role in driving the performance of the CNN. Tuning the learning rate yielded high and stable F1 scores across all models, indicating its key role in effective model training and convergence.

#### 4.1.2 Trade-off between training time and accuracy

Fig. 4 shows how an increase in test accuracy exhibits a plateau after a period of extended training, and may even decline. This illustrates the importance of careful hyperparameter tuning.

### 4.2 Two-factor analysis (optimizer interactions)

Two-factor analysis consists of changing two hyperparameters at the same time, with all others kept constant. This method reveals the interactions of two settings and their collective effect on model performance.

#### 4.2.1 Batch size × optimizer

The impact of varying the batch sizes is reflected differently in the optimizers across different architectures as shown in Fig. 5. In the case of MobileNetV2, SGD had a marginally higher performance than Adam at different batch sizes; but at larger batch sizes, the performance and results of Adam became unstable. On the contrary, for VGG16, Adam had better performance than SGD across all batch sizes. These findings show that interactions between optimizer choice and batch size are heavily reliant on the architecture used.

**Table 3** Spearman correlation between individual hyperparameters and the F1 score (one factor analysis)

| Architecture | Hyperparameter | Spearman $\rho$ | $p$-value |
|---|---|---|---|
| VGG16 | Learning rate | 0.566 | 0.006 |
| VGG16 | Batch size | −0.491 | 0.020 |
| VGG16 | Epochs | 0.488 | 0.021 |
| InceptionV3 | Batch size | −0.470 | 0.027 |
| ResNet50 | Learning rate | 0.433 | 0.044 |

#### 4.2.2 Learning rate × optimizer

The data illustrated in Fig. 6 shows that InceptionV3 experiences an optimizer performance peak at around 0.001. However, InceptionV3's SGD optimizer appeared to drop performance drastically around higher values. In the case of ResNet50, the recorded performance of the Adam optimizer showed no pronounced patterns over the examined learning rates, while the performance of SGD appeared to drop off rapidly as learning rates increased. Overall, these findings indicated that Adam is more sensitive to the learning rate than SGD.

### 4.3 Multi-factor optimization results

The multi-factor analysis conducted was to examine how several hyperparameters interact when changed together. This approach captures combined effects that do not appear in single-factor or two-factor tests, allowing us to identify configurations that remain consistently high-performing across different architectures.

#### 4.3.1 Interaction between learning rate, batch size, and dropout

The 3D scatter plot in Fig. 7 shows that the highest F1 scores were achieved with low learning rates and batch sizes up to 64, especially when dropout rates are moderate (0.1–0.3). As batch size or learning rate increased, F1 scores showed greater variability, with some combinations resulting in lower performance. Extremely high dropout rates or batch sizes tended to degrade performance. This suggests that consistent and reliable model performance is best achieved through balanced and less extreme hyperparameter settings.

#### 4.3.2 Three-factor hyperparameter optimization

The Optuna framework [35] was used to run RS and BO to jointly explore the learning rate, batch size, and dropout rate. The termination criterion used in both algorithms was fixed at 20 trial iterations maximum. Implementation of BO was based on the TPE sampler by Optuna, to probabilistically model the parameter space. Other factors were held constant in order to isolate these particular interactions. The results can be seen in Table 4.

The performance of both search methods was assessed using the test F1 score. Table 5 shows the best hyperparameter combinations obtained from the three-factor analysis. InceptionV3 achieved the highest F1 score (0.9249) using BO.

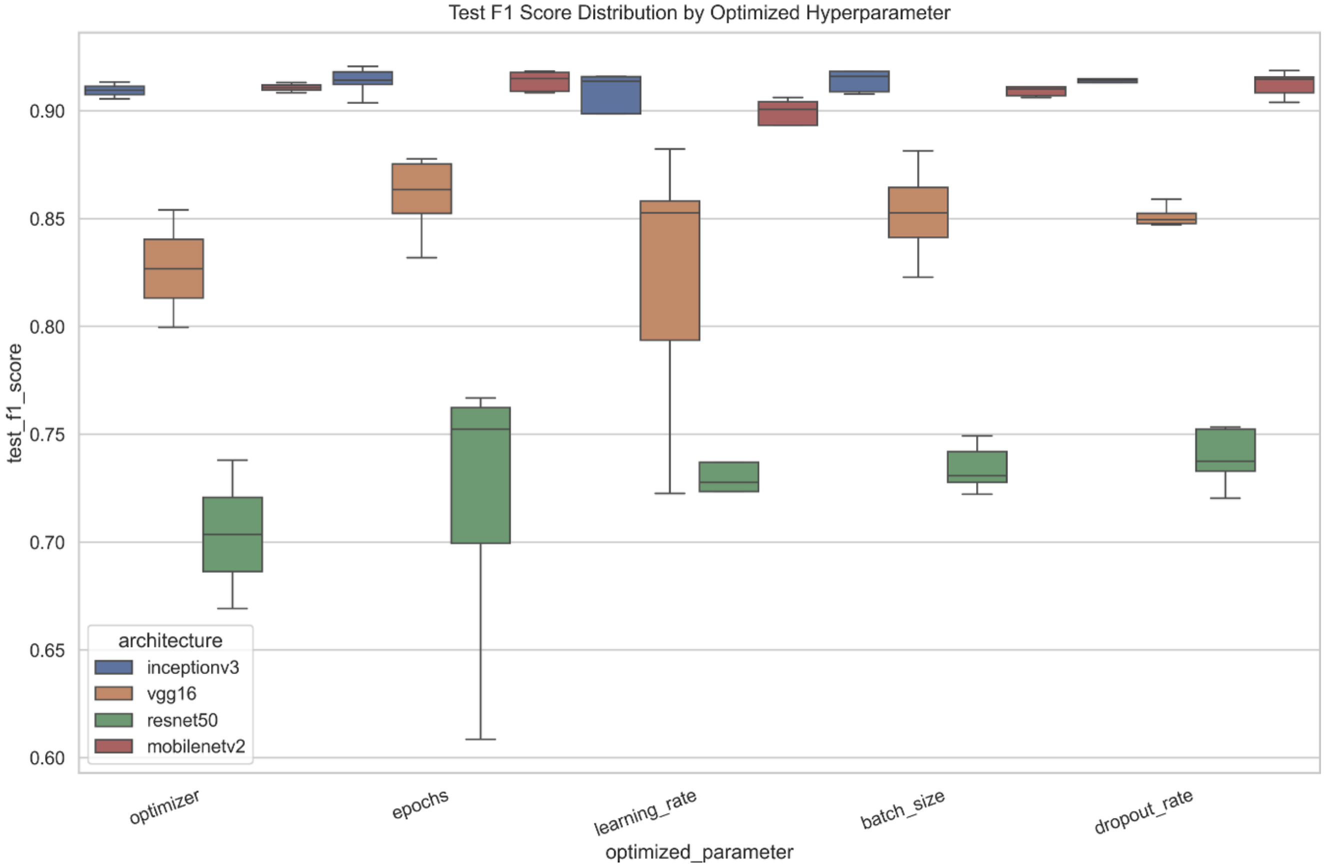


**Fig. 3** Boxplot showing learning rate and epochs as primary drivers of F1 score variance


**Fig. 4** Scatterplot illustrating diminishing returns in test accuracy with extended training time

### 4.3.3 Top 5 performing configurations across all five hyperparameters

Table 5 summarizes the top five configurations obtained from five-factor optimization. BO consistently achieved the highest F1 scores for the top models, with InceptionV3 reaching its best performance at a learning rate of 0.001, batch size of 64, and dropout of 0.2. Similarly, MobileNetV2 attained its best results under BO, typically using a larger

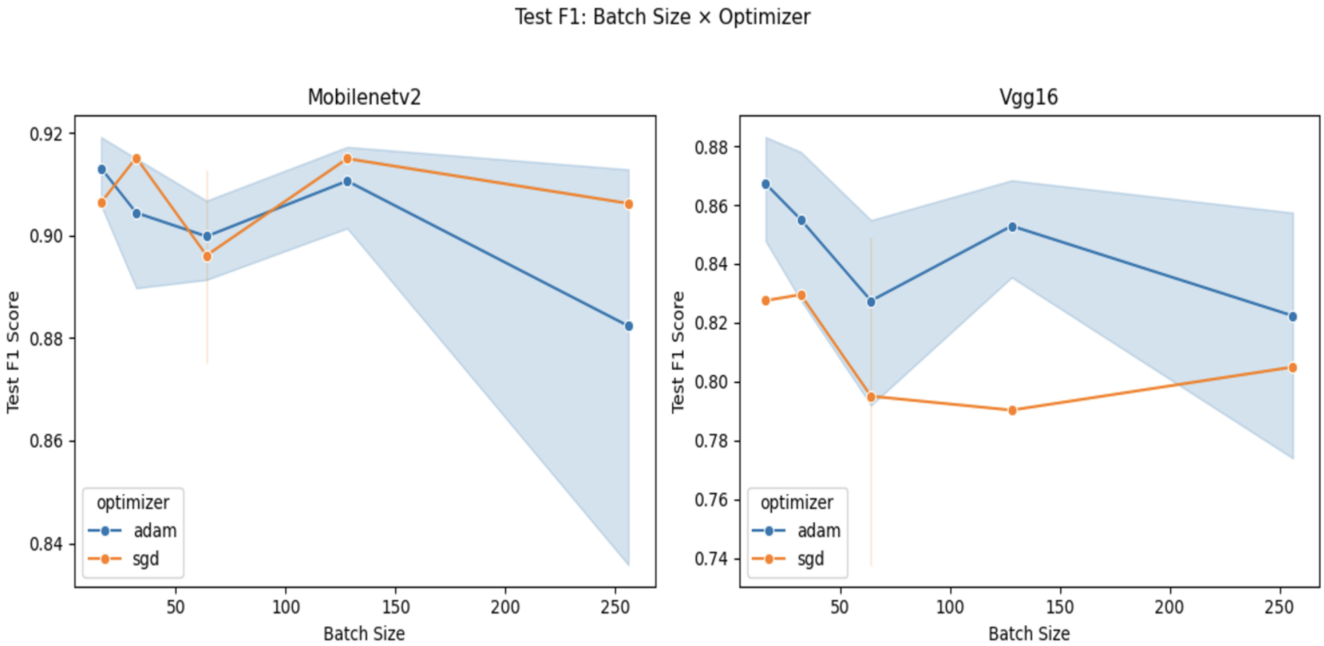


**Fig. 5** Line graph showing optimizer performance across different batch sizes for MobileNetV2 and VGG16

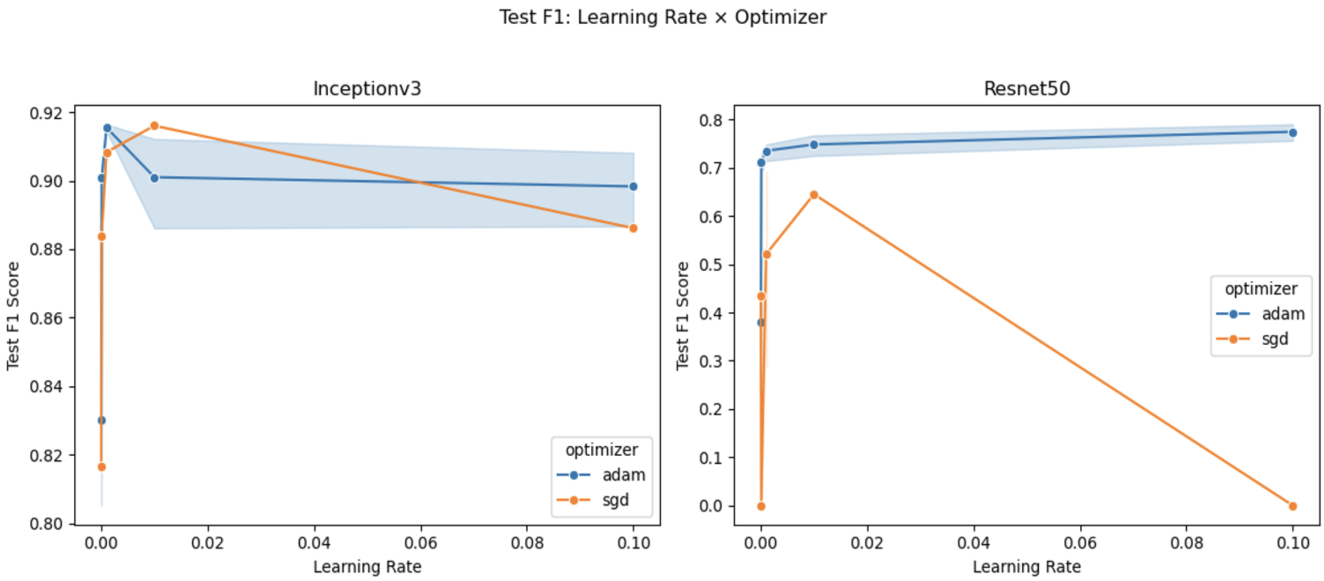


**Fig. 6** Line graph showing optimizer performance across different learning rates for InceptionV3 and ResNet50

batch size and higher dropout value. The RS configuration for MobileNetV2 performed slightly lower in comparison.

#### 4.3.4 Confusion matrix (CM) of the best model

The CM of the best InceptionV3 model (Fig. 8) achieved 420 true negatives, 30 false positives, 37 false negatives, and 413 true positives. According to this distribution, the model had a sensitivity (recall) of 91.8% and specificity of 93.3%. The model also had a precision of 93.2. These two clinical measures, along with the F1-score of 0.925, give a clearer and measurable baseline of the classification performance of the network on the test data.

### 4.4 Ablation study

The ablation study examined each hyperparameter (learning rate, batch size, dropout rate, number of epochs, and optimizer) separately and in conjunction with the final configured hyperparameters that were incorporated into the final CNN architectures.

Given the one-factor analysis, learning rate had the greatest influence on model performance. Higher learning rates (within the range of 0.001–0.01) were associated with higher F1 scores for VGG16 and ResNet50 (Spearman $\rho = 0.566$ and 0.434, $p < 0.05$). For InceptionV3 and MobileNetV2, learning rate also emerged as the most

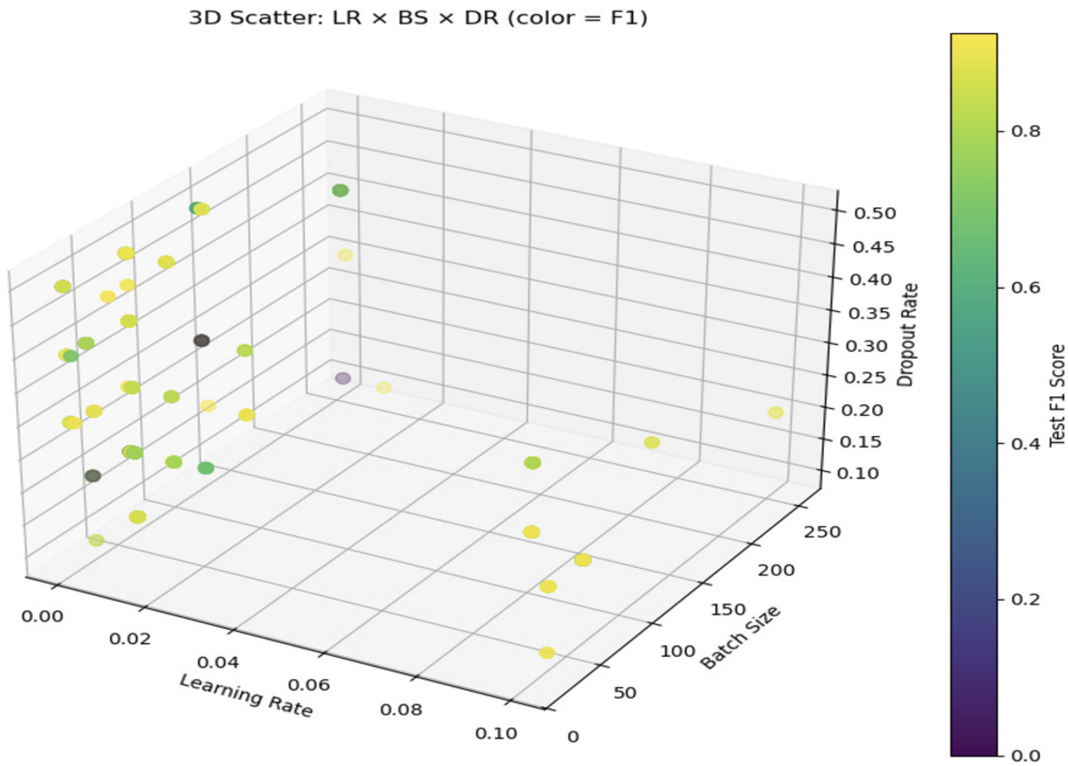


**Fig. 7** 3D scatter plot revealing peak F1 scores cluster at low learning rates with moderate batch sizes

**Table 4** Top hyperparameter configurations (3 factors)

| Algorithm | Architecture | Test F1 score | Learning rate | Batch size | Dropout rate | Epochs | Optimizer |
|---|---|---|---|---|---|---|---|
| BO | InceptionV3 | 0.9249 | 0.001 | 64 | 0.1 | 50 | Adam |
| BO | MobilenetV2 | 0.9201 | 0.001 | 32 | 0.3 | 20 | Adam |
| RS | InceptionV3 | 0.9234 | 0.001 | 64 | 0.4 | 40 | Adam |

**Table 5** Top 5 hyperparameter configurations (all the five factors)

| Algorithm | Architecture | Test F1 score | Learning rate | Batch size | Dropout rate | Epochs | Optimizer |
|---|---|---|---|---|---|---|---|
| BO | InceptionV3 | 0.9217 | 0.001 | 64 | 0.2 | 40 | Adam |
| BO | InceptionV3 | 0.9203 | 0.001 | 128 | 0.4 | 40 | Adam |
| BO | MobilenetV2 | 0.9194 | 0.001 | 128 | 0.4 | 30 | SGD |
| RS | MobilenetV2 | 0.9192 | 0.01 | 64 | 0.2 | 10 | SGD |
| BO | MobilenetV2 | 0.9189 | 0.01 | 128 | 0.4 | 20 | SGD |

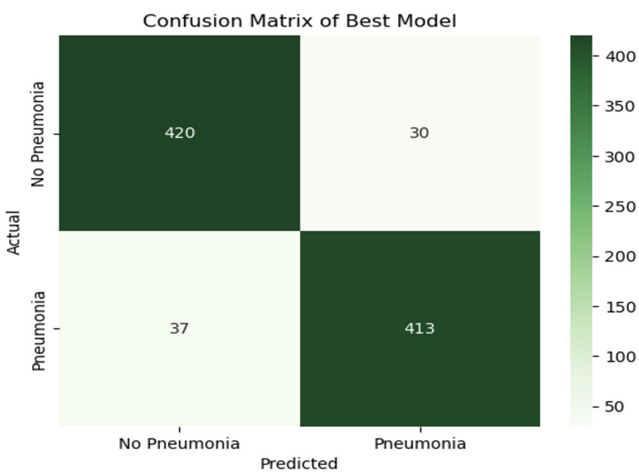


**Fig. 8** Confusion matrix of the best-performing model

influential hyperparameter according to Random Forest feature importance scores, which ranged from 0.325 to 0.741. In contrast, the impact of batch size was comparatively modest, as larger batch sizes slightly reduced performance in VGG16 and InceptionV3, whereas MobileNetV2 maintained good performance across a wide range of batch sizes. Most models did not have considerable dropout rate effects, except MobileNetV2, where a moderate dropout rate (0.1–0.3) preserved F1 score stability. The performance of InceptionV3 and MobileNetV2 was influenced positively by increasing the epoch count, even though VGG16 overfitted due to too many epochs. The Adam optimizer performed better than SGD across most architectures, particularly with extended training and moderate learning rates.

#### 4.4.1 Interaction effects

Two-factor and multi-factor analyses provided useful insights. In the case of learning rate and batch size, optimal performance was observed with learning rates between 0.001 and 0.01 and mid-sized batch sizes ranging from 16 to 128. In the analysis of learning rate and dropout rate, InceptionV3 showed minimal interactions between these parameters, but ResNet50 needed some tuning of both for optimal results. With respect to epochs and optimizer, models trained with Adam benefited more from prolonged training at low learning rates, compared to those using SGD. In the overall multi-factor interactions, it was evident that BO picked out configurations with high performance, which serves to indicate the importance of the combination of learning rate, batch size, dropout, epochs and optimizer in achieving the best F1 score.

The final hyperparameter adjustments include a learning rate of 0.001, which produced high F1 scores in every architecture, a batch size of 64 for a good trade-off between training performance and training time, a dropout rate of 0.1, enough to avoid overfitting without compromising performance, 40 epochs, permitting convergence without overfitting, and the Adam optimizer, which delivered the most consistent performance across architectures and hyperparameter combinations.

These settings were selected because they produced the highest F1 score of 0.9249 for InceptionV3. As shown in Table 2, the respective training times were relatively optimal, which reflects an optimal trade-off of about 0.4 h per experiment. Hence, this configuration was the optimal balance between accuracy, stability, and computational cost.

## 5 Discussion

This study demonstrated the critical role of hyperparameter tuning in CNNs, particularly for pneumonia detection. Out of all the hyperparameters included in the study, learning rate was found to be the most influential factor. Of the optimizers tested, Adam provided a solid platform for learning, particularly when combined with low learning rates.

Operation boundaries were moderately and clearly defined by batch size and dropout rate. Smaller batches (16 to 64) had a better F1 scores at a cost of increased computation time. Similarly, regularization requires dropout but a dropout rate above 0.2 reduced convergence and maximum accuracy, which shows that the optimal dropout to use in pneumonia classification is between 0.1 and 0.2.

The results also exemplify the trade-offs between computational cost and performance. Higher batch sizes result in smaller training time but may lead to reduced F1 score. Lower batch sizes are generally preferred as they lead to better performance, albeit at increased computational demand. Higher dropout rates, while effective at combating overfitting, tend to lower the maximum possible F1 score, as well as slow convergence. The results clearly indicate that both excessively large and small learning rates negatively affect training stability and convergence. The final hyperparameter configuration resulted in a strong balance between accuracy, training performance, and computation time.

The optimized InceptionV3 architecture reached a test F1 score of 0.925, outperforming the results (87.9%) reported by Saboo et al. [36] and approximating those of Pechnikov et al. [17]. Although Saboo et al. [36] reported positive effects of using adaptive optimizers, the current research confirms that Adam optimizer, in combination with a strictly limited dropout rate of 0.1, provides significantly superior classification accuracy in the case of adult datasets.

These results have important practical implications, as they can assist radiologists in classifying pneumonia cases rapidly and accurately. The findings in this study indicate that merely selecting a standard architecture is not sufficient to achieve high performance. Careful tuning based on dataset type is important. The model offers great potential to be incorporated into clinical pathways due to its high sensitivity and specificity in helping to minimize prioritization errors and misdiagnoses.

## 6 Conclusion

Hyperparameter tuning is critical for improving the accuracy and performance of CNNs in pneumonia diagnosis from chest X-ray images. The InceptionV3 model with a learning rate of 0.001, batch size of 64, dropout rate of 0.1, and trained for 40 epochs using the Adam optimizer, showed optimized accuracy with an F1 score of 0.925. The deployment of AI models in medicine provides useful tools for radiologists to fully integrate AI into their practices and expedite pneumonia diagnosis.

Even though the use of these optimized AI models can give radiologists useful diagnostic information, they cannot fully realize their full clinical potential without integrating model interpretability. The decision-making procedure of CNNs must be made transparent to medical professionals in order to develop confidence in these systems.

## 7 Limitations and future work

One of the main weaknesses of this study is that it uses only one dataset, which prevents the comparison of the model with different patient demographics. The existing data distribution is extremely biased towards adult groups, i.e., the fitted hyperparameters will not perfectly apply to pediatric patients.

Moreover, given that the process of data division was done on an image-by-image basis but not on a patient-by-patient basis, two or more radiographs of the same patient can be found on the training and testing sets.

External validation should take precedence in future work to tackle the limitations inherent in the use of single datasets. The optimized hyperparameter settings should be tested using independent X-ray data that are acquired from other hospitals to determine whether they are diagnostic in real clinical setting.

Scalability is another consideration because training deep learning models like InceptionV3 is resource- and time-intensive. Future research should focus on the use of hyperparameter tuning and less resource-intensive CNN architectures that train faster.

Developing model interpretability and explainability tools should be a key focus of future work to build clinical trust and practical utility. Methods like gradient-weighted Class Activation Mapping (Grad-CAM) should be included to visually indicate the radiographic parts informing the model to make its judgments. This would enable clinicians to independently verify the AI diagnostic logic.

**References**

[1] Reyes, L. F., Conway Morris, A., Serrano-Mayorga, C., Derde, L. P. G., Dickson, R. P., Martin-Loeches, I. "Community-acquired pneumonia", The Lancet, 406(10517), pp. 2371–2388, 2025. https://doi.org/10.1016/S0140-6736(25)01493-X

[2] UNICEF "Pneumonia in children: What you need to know", [online] Available at: https://www.unicef.org/stories/childhood-pneumonia-explained [Accessed: 05 January 2026]

[3] Kopf, J. "Pneumonia: Everything You Need to Know", HealthCentral, 11 May 2021. [online] Available at: https://www.healthcentral.com/condition/pneumonia [Accessed: 05 January 2026]

[4] Makhnevich, A., Sinvani, L., Feldhamer, K. H., Zhang, M., Richardson, S., McGinn, T. G., Cohen, S. L. "Comparison of Chest Radiograph Impressions for Diagnosing Pneumonia: Accounting for Categories of Language Certainty", Journal of the American College of Radiology, 19(10), pp. 1130–1137, 2022. https://doi.org/10.1016/j.jacr.2022.05.020

[5] Goyal, D., Inada-Kim, M., Mansab, F., Iqbal, A., McKinstry, B., …, Burke, D. "Improving the early identification of COVID-19 pneumonia: A narrative review", BMJ Open Respiratory Research, 8(1), e000911, 2021. https://doi.org/10.1136/bmjresp-2021-000911

[6] Ma, Y., Fan, S., Xi, J. "Recent updates regarding the management and treatment of pneumonia in pediatric patients: a comprehensive review", Infection, 53(6), pp. 2341–2359, 2025. https://doi.org/10.1007/s15010-025-02605-w

[7] Gupta, J., Joshi, M. L., Gupta, N. K. "Exploring Chest Disease Classification Methods Using X-ray Image Analysis", Austin Journal of Radiology, 12(1), 1248, 2025. [online] Available at: https://austinpublishinggroup.com/radiology/fulltext/ajr-v12-id1248.pdf [Accessed: 05 January 2026]

[8] Sarvamangala, D. R., Kulkarni, R. V. "Convolutional neural networks in medical image understanding: a survey", Evolutionary Intelligence, 15(1), pp. 1–22, 2022. https://doi.org/10.1007/s12065-020-00540-3

[9] Mousavi, M., Hosseini, S. "A deep convolutional neural network approach using medical image classification", BMC Medical Informatics and Decision Making, 24(1), 239, 2024. https://doi.org/10.1186/s12911-024-02646-5

[10] Mathi, S., Deb, D., Chaudhuri, A. K., Joseph, L., Biswas, S. "A Review Of Convolutional Neural Networks For Medical Image Analysis: Trends And Future Directions", Journal of Neonatal Surgery, 14(7), pp. 90–97, 2025. https://doi.org/10.63682/jns.v14i7.5117

[11] Mendes, J., Lima, J., Costa, L., Hendrix, E. M. T., Pereira, A. I. "Impact of hyper-parameter tuning on CNN accuracy in agricultural image classification", Smart Agricultural Technology, 11, 101016, 2025. https://doi.org/10.1016/j.atech.2025.101016

[12] Lacerda, P., Barros, B., Albuquerque, C., Conci, A. "Hyperparameter Optimization for COVID-19 Pneumonia Diagnosis Based on Chest CT", Sensors, 21(6), 2174, 2021. https://doi.org/10.3390/s21062174

[13] Gea, M. N., Wanayumini, W., Rosnelly, R. "Impact of Hyper-parameter Tuning on CNN-Based Algorithm for MRI Brain Tumor Classification", Jurnal Teknik Informatika, 18(1), pp. 87–100, 2025. https://doi.org/10.15408/jti.v18i1.44147

[14] Shahira, F., Negara, B. S. "Lung X-Ray Image Classification Using DenseNet-169 and Bayesian Optimization", Knowbase: International Journal of Knowledge in Database, 5(1), pp. 18–27, 2025. https://doi.org/10.30983/knowbase.v5i1.9618

[15] Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., …, Ng, A. Y. "CheXNet:Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning", [preprint] arXiv, arXiv:1711.05225, 25 December 2017. https://doi.org/10.48550/arXiv.1711.05225

[16] Ayan, E., Ünver, H. M. "Diagnosis of Pneumonia from Chest X-Ray Images Using Deep Learning", In: 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), Istanbul, Türkiye, 2019, pp. 1–5. ISBN 978-1-7281-1013-4 https://doi.org/10.1109/EBBT.2019.8741582

[17] Pechnikov, A., Bogdanov, N., Nwohiri, A., Nwohiri, I. "Two Approaches for Detecting Pneumonia from Chest X-ray Images: Neural Network vs Kolmogorov Complexity", Periodica Polytechnica Electrical Engineering and Computer Science, 67(3), pp. 345–354, 2023.
https://doi.org/10.3311/PPee.21616

[18] Kermany, D. S., Goldbaum, M., Cai, W., Valentim, C. C. S., Liang, H., …, Zhang, K. "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning", Cell, 172(5), pp. 1122–1131, 2018.
https://doi.org/10.1016/j.cell.2018.02.010

[19] Brima, Y., Atemkeng, M., Tankio Djiokap, S., Ebiele, J., Tchakounté, F. "Transfer Learning for the Detection and Diagnosis of Types of Pneumonia including Pneumonia Induced by COVID-19 from Chest X-ray Images", Diagnostics, 11(8), 1480, 2021.
https://doi.org/10.3390/diagnostics11081480

[20] Chouhan, V., Singh, S. K., Khamparia, A., Gupta, D., Tiwari, P., Moreira, C., Damaševičius, R., de Albuquerque, V. H. C. "A Novel Transfer Learning Based Approach for Pneumonia Detection in Chest X-ray Images", Applied Sciences, 10(2), 559, 2020.
https://doi.org/10.3390/app10020559

[21] Liao, L., Li, H., Shang, W., Ma, L. "An Empirical Study of the Impact of Hyperparameter Tuning and Model Optimization on the Performance Properties of Deep Neural Networks", ACM Transactions on Software Engineering and Methodology (TOSEM), 31(3), 53, 2022.
https://doi.org/10.1145/3506695

[22] Ha, D., Carstensen, J. "Automatic hyperparameter tuning of topology optimization algorithms using surrogate optimization", Structural and Multidisciplinary Optimization, 67(9), 157, 2024.
https://doi.org/10.1007/s00158-024-03850-7

[23] Wojciuk, M., Swiderska-Chadaj, Z., Siwek, K., Gertych, A. "Improving classification accuracy of fine-tuned CNN models: Impact of hyperparameter optimization", Heliyon, 10(5), e26586, 2024.
https://doi.org/10.1016/j.heliyon.2024.e26586

[24] El-Hassani, F. Z., Amri, M., Joudar, N.-E., Haddouch, K. "A New Optimization Model for MLP Hyperparameter Tuning: Modeling and Resolution by Real-Coded Genetic Algorithm", Neural Processing Letters, 56(2), 105, 2024.
https://doi.org/10.1007/s11063-024-11578-0

[25] Yilmaz, A., Kuş, İ. "General CNN model for biomedical image classification via genetic algorithm-based hyperparameter optimization", Ain Shams Engineering Journal, 17(1), 103891, 2026.
https://doi.org/10.1016/j.asej.2025.103891

[26] Awotunde, J. B., Imoize, A. L., Ayoade, O. B., Abiodun, M. K., Do, D.-T., Silva, A., Sur, S. N. "An Enhanced Hyper-Parameter Optimization of a Convolutional Neural Network Model for Leukemia Cancer Diagnosis in a Smart Healthcare System", Sensors, 22(24), 9689, 2022.
https://doi.org/10.3390/s22249689

[27] Raiaan, M. A. K., Sakib, S., Fahad, N. M., Mamun, A. A., Rahman, M. A., Shatabda, S., Mukta, M. S. H. "A systematic review of hyperparameter optimization techniques in Convolutional Neural Networks", Decision Analytics Journal, 11, 100470, 2024.
https://doi.org/10.1016/j.dajour.2024.100470

[28] Aguerchi, K., Jabrane, Y., Habba, M., El Hassani, A. H. "A CNN Hyperparameters Optimization Based on Particle Swarm Optimization for Mammography Breast Cancer Classification", Journal of Imaging, 10(2), 30, 2024.
https://doi.org/10.3390/jimaging10020030

[29] Ahamed, M. F., Nahiduzzaman, M., Mahmud, G., Shafi, F. B., Ayari, M. A., Khandakar, A., Abdullah-Al-Wadud, M., Islam, S. M. R. "Improving Malaria diagnosis through interpretable customized CNNs architectures", Scientific Reports, 15(1), 6484, 2025.
https://doi.org/10.1038/s41598-025-90851-1

[30] Asiri, A. A., Shaf, A., Ali, T., Aamir, M., Irfan, M., Alqahtani, S. "Enhancing brain tumor diagnosis: An optimized CNN hyper-parameter model for improved accuracy and reliability", PeerJ Computer Science, 10, e1878, 2024.
https://doi.org/10.7717/peerj-cs.1878

[31] Sumona, R. B., Biswas, J. P., Shafkat, A., Rahman, M. M., Faruk, M. O., Majeed, Y. "An integrated deep learning approach for enhancing brain tumor diagnosis"", Healthcare Analytics, 8, 100421, 2025.
https://doi.org/10.1016/j.health.2025.100421

[32] Python Software Foundation "Python, (3.14.4)", [computer program] Available at: https://www.python.org/ [Accessed: 20 December 2025]

[33] NumPy team "NumPy, (2.4.4)", [computer program] Available at: https://numpy.org/ [Accessed: 20 December 2025]

[34] Zenodo "TensorFlow, (2.21.0)", [computer program]
https://doi.org/10.5281/zenodo.4724125

[35] Preferred Networks, Inc. "Optuna, (v4.8.0)", [computer program] Available at: https://optuna.org/ [Accessed: 22 December 2025]

[36] Saboo, Y. S., Kapse, S., Prasanna, P. "Convolutional Neural Networks (CNNs) for Pneumonia Classification on Pediatric Chest Radiographs", Cureus, 15(8), e44130, 2023.
https://doi.org/10.7759/cureus.44130