LAUTECH Journal of Engineering and Technology 18 (2) 2024: 182-192
10.36108/laujet/4202.81.0271

Optimized convolution neural network-based model for detection and classification of pulmonary diseases

[1]*Olajide A. T., [2]Babatunde R. S., [2]Abdulsalam S. O., [2]Ajao, J. F. [2] Isiaka R. M. [1]Ayeni J. K.

[1]*Department of Computer Science, Institute of Information & Communication Technology, Kwara State Polytechnic, Ilorin, Kwara State. Nigeria.*
[2]*Department of Computer Science, Faculty of Information and Communication Technology, Kwara State University (KWASU), Malete, Nigeria.*

Article Info

Article history:

Received: March 22, 2024
Revised: May 25, 2024
Accepted: June 15, 2024

Keywords:

Chest X-ray,
Deep Learning,
Convolution neural network,
Data augmentation,
Pelicanoptimization,
pulmonary disease.

Corresponding Author:
olajide.an@kwarastatepolytechnic.edu.ng.

ABSTRACT

Pelican Optimization Algorithm-based Convolutional Neural Network (POA-CNN) method for the automated identification of pulmonary disorders such as COVID-19 and pneumonia is proposed in this research. The study aims to enhance the efficiency of CNN models in diagnosing lung diseases by using the Pelican Optimization Algorithm (POA), and by addressing drawbacks like a lack of flexibility in hyperparameter modifications. The three primary phases of the model are feature extraction via POA-based hyperparameter optimization, image classification, and image pre-processing. This approach improves existing systems' performance in detecting pulmonary diseases, highlighting the potential of deep learning in identifying and categorizing human diseases. The study uses resizing, grayscale, and augmentation methods to optimize an existing CNN model. A Convolutional Neural Network (CNN) is then applied to classify Pneumonia and COVID-19 cases. The proposed model achieves an accuracy rate of 97.28% and 97.00%, outperforming existing models. This technique is effective in detecting and classifying other pulmonary diseases and can be used to automatically detect and classify these diseases. Higher accuracy findings show how successful the model is, making it a useful tool for pulmonary illness identification.

INTRODUCTION

CNN is a deep learning (DL) technique that has proven to be successful in resolving a range of image-related issues (Hashmi *et al*., 2020). The CNN procedure is separated into two parts: the fully connected layer and the feature extraction layer. To extract features from an image, the feature extraction layer uses polling and convolution techniques (Jain *et al*., 2020). The multidimensional array, which is the feature map from the feature extraction layer, needs to be flattened or reshaped into a vector to be used as input for the fully linked layer. This layer, which consists of input, hidden, activation function, and output layers, functions similarly to an Artificial Neural Network (ANN) (Apostolopoulos and Mpesiana, 2020). The goal of the research is to improve the effectiveness of cutting-edge CNN models in the accurate and efficient diagnosis of lung diseases, which is essential for early identification and lower patient death rates. This study aims to fill the existing gap by employing the Pelican Optimization Algorithm (POA) to enhance the performance of the CNN network. The paper presents the POA, which optimizes CNN hyperparameters. Based on its ability to explore and exploit both locally and globally, POA can optimize the CNN architecture and increase CNN accuracy by identifying a better configuration.

Hyperparameter tuning is necessary as the default settings of a CNN-based model cannot guarantee its optimal performance (Yang and Shami 2021). The process of finding a model's ideal hyperparameter values, known as hyperparameter tuning, necessitates choosing the best optimization strategy to take local values into account (Yang and Shami, 2021: Shekar and Dagnew, 2019). A metaheuristic algorithm is a collection of techniques for dealing with challenging problems related to search space optimization, such as hyperparameter adjustment (Jain *et al*., 2020, Sahedi *et al*., 2021). One popular kind of conventional optimization technique that can adjust hyperparameter tuning by figuring out its gradient is a metaheuristic algorithm. In addition, conventional techniques for hyperparameter optimization include Grid Search (GS) and Random Search (RS) (Bharati *et al*., 2021). Traditional optimization techniques are less effective in hyperparameter tuning problems compared to alternative strategies like metaheuristic algorithms (Zahedi *et al.*, 2021, Rajagopal *et al*., 2023).A metaheuristic algorithm is a set of techniques used to solve complex and large-scale optimization problems, such as hyperparameter tuning in search space optimization. The collection of hyperparameter combinations that the Genetic Algorithm (GA) discovers to be effective in each generation is then iterated through until the combination with the best performance is identified (Alibrahim *et al.*, 2021, Zahedi *et al*., 2021). To find and update the global ideal throughout each iteration until the final optimal value is obtained, the Particle Swarm Optimization (PSO) algorithm communicates with other particles (Alibrahim *et al.*, 2021).

Inspired by the hunting habit of pelicans, the POA is an evolutionary algorithm that finds optimum solutions quickly by exploring search regions (Trojovsky and Dehghani, 2022). Because complex processes like permutations are not included; it is less complex than GA and PSO and provides more consistent performance when compared with other optimization techniques. POA's strong efficiency makes it perfect for hyperparameter tuning problems with huge configuration spaces. It may be applied with several hyperparameters in DL. Though it has benefits over other metaheuristic optimization methods; it also confronts challenges such as premature convergence, an imbalance between exploration and exploitation, and a lack of population variety (Trojovsky and Dehghani, 2022). Using the five created hyperparameters, adjust the hyperparameters on the fully connected layer procedure to obtain the optimal minimal error. The number of hidden layers, the kind of activation function, the loss function, the number of epochs, and the batch sizes are the five hyperparameters. The use of POA as a CNN tuning hyperparameter, which reduces long experimental and computation times and automatically creates efficient architectures without requiring extensive testing on different CNN parameter combinations, (Trojovsky and Dehghani, 2022), is the innovative aspect of this work. The following are the study's specific contributions:

Firstly, using POA, the hyperparameters of the current CNN are tuned to find the most accurate results in the diagnosis of pneumonia and COVID-19 from chest X-ray pictures. A CNN-POA model with 97.28% and 97.00% accuracy rates is suggested for the automated identification of pneumonia and COVID-19 illnesses respectively, utilizing chest X-ray images. The evaluation in terms of different performance metrics such as accuracy, sensitivity, specificity and precision were considered.

Hashmi *et al.*(2020) developed a model for detecting pneumonia in chest X-ray images. The model's final prediction was achieved by integrating the predictions of pre-trained deep learning models (Xception, Inception V3, ResNet 18, Mobile Net V3, and Dense Net 121).The suggested model achieves 96.43% accuracy and outperforms the other stand-alone models. Because the models used had very complicated architectures, it is necessary to develop a model in which the weights corresponding to many models are effectively calculated. Jain *et al.*, (2020) presented six CNN models to classify chest X-ray pictures into pneumonia and non-pneumonia categories. These models differ in how many hyper-factors, convolutional layers, and other parameters are employed. The pre-trained models (VGG16, VGG19, Inception V3, and ResNet 50) have accuracy values of 87.28%, 88.46%, 70.99%, and 77.56%, respectively, whereas the first and second models, each with three Convolutional layers, have accuracy values of 85.26% and 92.31%, respectively. The models under discussion here focus primarily on the recall metric as a performance measure for lowering the number of false negatives. With a 97% recall rate, the second model performed the best. However, to improve classification accuracy, these models require adjustments to every parameter and hyper-parameter.

Using Deep Learning, Bharati *et al.*, (2021) created an optimized residual network (CO-ResNet) that improved the conventional ResNet 101 by hyperparameter tweaking. The experimental assessment showed a COVID-19 detection rate accuracy of 96.74%, with 92.08% and 91.32% for healthy lungs, and 83.68% for normal, healthy lungs with ResNet152 as the foundation. The small size of the dataset used in this investigation was a drawback.

Buvana *et al.*, (2021) introduced the Black Widow Optimization (BWO) method (BWO-CNN), a distinctive COVID-19 diagnostic and classification model. The proposed BWO-CNN model recognizes and classifies COVID-19 using an image processing technique. It modifies CNN's hyperparameters via the Black Widow Optimization (BWO) method. Ultimately, the Extreme Learning Machine with Autoencoder (ELM-AE) classifier was used to identify and categorize COVID-19 under a variety of class names. The suggested BWO-CNN model was experimentally verified, and the outcomes were contrasted with those of earlier methods. The test set was too small to yield results that would be statistically significant, which reduced the generalizability of the model. In order to automatically create a CNN model for an automated skin lesion classification system that is trustworthy, durable, and accurate for early skin lesion diagnosis, (Salih and Duffy, 2023) use a genetic method. Utilizing four publicly available datasets, the refined CNN model is trained to identify anomalies based on characteristics of skin lesions in various orientations. The precision measure and F-score are all the best scoring for the model. Compared to other current approaches, their scores compare favourably.

ALGORITHM FOR POA

This paper presents the POA. It has the advantages of fast convergence and simple computing. Inspired by the intriguing foraging activities of pelicans, the Pelican Optimization method is a naturalistic optimization method Trojovsky and Dehghani, (2022). These birds can grab their prey and remove the water using their lengthy beaks and enormous throat pouches. They hunt both alone and in groups, and they live in groups. Reptiles, fish, crabs, and turtles are some of the things that pelican birds eat. In case of hunger, they would consume marine food as well. After finding its prey, the pelican dives quickly into the water to seize it. Pelicans, at low tide, pursue fish in shallow water, allowing them to be easily captured by spreading their wings across the water's surface. A pelican's mouth becomes too filled with water when it catches its prey. The pelican tilts its head forward before consuming the food to expel the excess water. The POA aims to replicate pelican hunting behaviour, with a mathematical formulation and POA pseudo-code provided for easy explanation.

Initialization

Equation (1) is used to randomly initialize the population's members, displaying the upper and lower bounds of the problem variables. Each person contributes values for the variables in the optimization problem based on where they are in the search space.

$$Y_{p,q} = L_q + (U_q - L_q); p = 1,2,3, \ldots \ldots n, q = 1,2,3, \ldots \ldots \ldots r \quad (1)$$

$Y_{p,q}$ represents the value of the qth variable for the pth candidate solution, it suggests a random value between 0 and 1. The terms L_q and U_q represent the lower and upper bound values of the qth variable, n represents the size of the population, and r stands for the total number of issue variables. Each potential solution's fitness functions are assessed once the population members have been initialized. The following equation illustrates how a vector function, also known as an objective function vector, is used to calculate the fitness function values.

$$J = \begin{bmatrix} J_1 \\ J_p \\ J_n \end{bmatrix} nx1 = \begin{bmatrix} J(Y_1) \\ J(Y_p) \\ J(Y_n) \end{bmatrix} nx1 \quad (2)$$

While the fitness function is represented as J, the fitness function of the pth Candidate solution is written as J_p.

Phase of Exploration

The Pelicans, who make up the population, locate the prey and enter that area of the search arena during the exploration phase. One of the main features of this POA is the arbitrary production of the prey's position from the search dimension. When optimization problems are solved, it increases the POA's capacity for exploration. The exploration process may be represented quantitatively as follows:

$$Y^1{}_{p,q} = \begin{cases} Y_{p,q} + .\left(P_q - N.Y_{p,q}\right), J_V < J_P \\ Y_{p,q} + .\left(Y_{p,q} - P_q\right), \quad \text{otherwise} \end{cases} \quad (3)$$

The POA uses a random integer N, prey location $Y^1{}_{P,q}$ and fitness function value j_v to determine a pelican's new position at the qth dimension of the exploration phase. If N is set to 2, the pelican will be more displaced, affecting its exploration capability. A higher fitness function value prevents the pelican from migrating to the worst possible location, making its new location acceptable. This procedure is mathematically represented as follows:

$$Y_p = \begin{cases} Y^1{}_p, j^1 p < j_P \\ Y_{p,} \text{otherwise} \end{cases} \quad (4)$$

The pelican's fitness function value is displayed by $j^1{}_p$, and its new position is given by the parameter $Y^1{}_p$.

Exploitation Phase

During the exploitation phase, pelicans flap their wings over the water's surface to encourage the fish to swim upward. Pelicans gather fish in their large throat pouches upon their surface. By assisting pelicans in obtaining more fish from the hunting region, this mechanism improves the Pelican Optimization Algorithm's capacity for both exploration and exploitation. Here is a description of this strategy's numerical simulation:

$$Y^2{}_{p,q} = Z_{p,q} + r.\left(1 - {}^{T}/_{S}\right).(2.-1).Y_{p,q} \quad (5)$$

The new location of the pth pelican at the qth dimension of the exploitation phase is shown by the equation (5), which also contains a constant of r = 0.2, a neighborhood radius of $Y_{p,q,}$ the number of iterations (T), and the maximum iteration (S). The new position of the pelican is also included in the updated calculation.

$$Y_p = \begin{cases} Y^2{}_p j^2 p \ < j_P \\ Y_{p,}\text{otherwise} \end{cases} \quad (6)$$

The values $Y^2{}_p$ and $j^2{}_p$, respectively, reflect the pth pelican's new position and fitness function value.

Algorithm 1: Pseudo-code of POA

Start POA

1. Provide the specifics of the optimization issue.
2. Determine the number of iterations (T) and the population size of POA (P).
3. The objective function is computed and the position of the pelicans is initialized.
4. In the case of t = 1:T
5. Generate prey location randomly.
6. In the case when I = 1:P
7. Phase 1: Approaching the prey (the exploring phase).
8. For j = 1:m
9. Applying (3) to determine the dimension state of the jth
10. End.
11. Using (4), update the ith person in the population.
12. Phase 2: Flying above the surface of the water (exploitation).
13. Considering that j = 1:m
14. Applying (6) to determine the dimension state of the jth
15. End.
16. Using (6), update the ith person in the population.
17. End
18. Update the top contender solution.
19. End.
20. The output should be the POA's best candidate answer.
21. Put an end to POA.

MATERIALS AND METHODS

A framework of the planned study's classification and identification procedures for lung disorders is given in this section. The section covers the methods utilized, the dataset that was used, and the pre-processing using data augmentation approaches. The procedure for the recommended approach is shown in Figure 1.

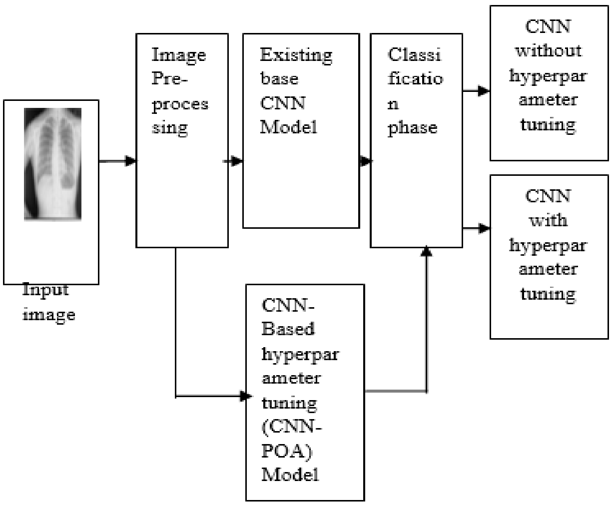

Figure 1: CNN-POA Model Framework

DATASET

The dataset utilized in this study, which includes COVID-19 picture and chest X-ray scan of pneumonia as seen in Figures 2 and 3, was gathered (downloaded) from the Kaggle and github data repositories. The chest X-ray pictures are pre-processed to remove any blurry or low-quality scans. The generated original images collated is 1,700.

Figure 2: Pneumonia

Figure 3: Covid-19

Chest X-ray images (source: Kaggle's Chest X-ray et 2020)

Pre-Processing Images datas

and Data Augmentation

There are several resolutions of the photographs in the dataset. However, the CNN models usually need images that have specific dimensions. The datasets were all reduced to a set size as smaller input images allow for quicker image processing, which speeds up the model for that specific connected job. Data Augmentation is a popular strategy used to significantly increase the number of training data by adding small-picture changes to each training period. In this piece, the variations that are employed are grayscale, picture rotation, and horizontal and vertical flipping. After pre-processing procedures, the number of obtained augmented images is 1,316. Afterwards, all datasets were randomly split into two independent datasets, with 70% and 30% used for training and testing, respectively. The generated original and augmented images (3,016) are input into the proposed model for training and testing. The CNN model can learn from a larger amount of data than was originally in the dataset, hence this method is essential to getting high accuracy.

State-of-the-art CNN MODEL

The output layer, which comes in last, is responsible for giving the input data the required label. Using the CNN technique, individuals with pneumonia and COVID-19 are classified.

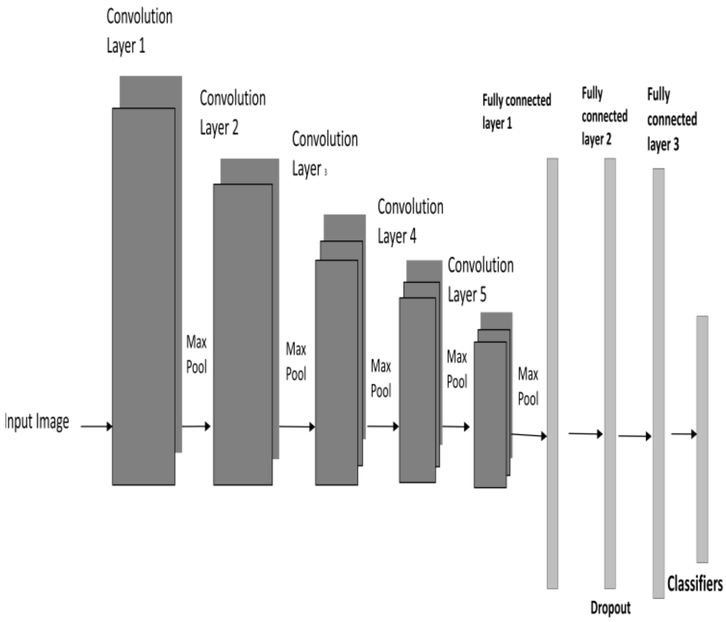

Figure 4. CNN Model Architecture (Bharati et al., 2021)

THE PROPOSED MODEL

Convolutional Neural Networks (CNNs) with POA-based hyper-parameter tweaking are used in this study. Three primary phases comprise the CNN-POA technique: pre-processing of the images, features extraction via POA-based hyperparameter tuning, and picture classification through the use of a (CNN). The model uses different layers to classify chest X-rays as either pneumonia or COVID-19. Fig. 5 displays the whole process of the CNN-POA system. Important components of the proposed model include convolution, maxpooling, flattening, sigmoid function, Rectified Linear Unit (RELU) activation function, and fully connected layer (FC), each of which performs a different purpose.The POA-CNN model is used for feature extraction after the input picture has undergone preprocessing. The use of POA feature extraction aids in the first CNN hyperparameter selection process.

The fundamental layer of CNNs in neural networks is called the convolution layer. It completes the main task of extracting specific information from the incoming images. This layer applies a filter to the input picture, and the resulting values are what make up the feature map. This layer slides across the pattern using kernels to extract both high-level and low-level information.

This work used max-pooling to summarize the feature maps derived from convolution operations by adjusting the parameters considered during the training operation. The max pooling algorithm uses the matrix size in each feature map to select the maximum value, reducing output neurons and regulating overfitting processes to reduce computation time and restrict network overfitting, thereby reducing overall performance. This layer flattens the input by transforming a two-dimensional feature matrix into a vector, which is then used to train a fully connected neural network classifier. The input image from the layers in Figure 4 is flattened and moved to the FC layer in this work. After that, the flattened vector passes through a few more FC levels, which is where the operations on the mathematical functions often take place. This is the moment where the classifying process begins.

The study uses the ReLU function, a sigmoid activation function, to convert the output of each convolution layer and extend the framework to include non-linearity. This function anticipates output pictures concurrently in the last layer of the fully connected layer, offering a larger gradient of convergence and lower processing cost. ReLU yields zero if the input is negative and equals data if the input is positive.

THE OPTIMIZATION OF HYPER-PARAMETER

The paper presents the POA-CNN methodology as a method for diagnosing and classifying COVID-19 and pneumonia. This methodology comprises preprocessing, feature extraction, parameter optimization, and classification stages. The optimization was performed using Pelican Optimization Algorithm (POA) to improve the performance of a cutting-edge Convolutional Neural Network (CNN) model for identifying lung illnesses using chest X-ray images.

POA was utilized to optimize key parameters of the Convolutional Neural Network (CNN) model, including the batch size, number of epochs, and learning rate. The objective was to enhance the accuracy, efficiency, and convergence of the CNN in its classification tasks. In the training phase, the batch size of 256 images was considered (i.e. within the bounds of 64, 128, and 256) and the number of epochs was set to 30 (within the search space of 10-30), The models were trained to perform binary-class classification using sigmoid activation in the output layer. In addition, the study enhanced the performance of the classifier section in CNN architecture models by implementing POA, resulting in improved accuracy in distinguishing between COVID-19 and pneumonia patients using chest X-ray pictures.

The optimization method aims to decrease the occurrence of incorrect positive results and improve the overall performance metrics of the CNN model. This resulted in enhanced sensitivity, specificity, precision, and accuracy in the diagnosis of lung-related disorders from chest X-ray pictures. The model employs distinct layers to categorize chest X-rays as either pneumonia or COVID-19. Figure 5 illustrates the complete procedure of the CNN-POA system. The suggested model consists of several crucial elements, namely convolution, max-pooling, flattens, sigmoid function, RELU activation function, and fully connected layer. Each of these components serves a distinct purpose.

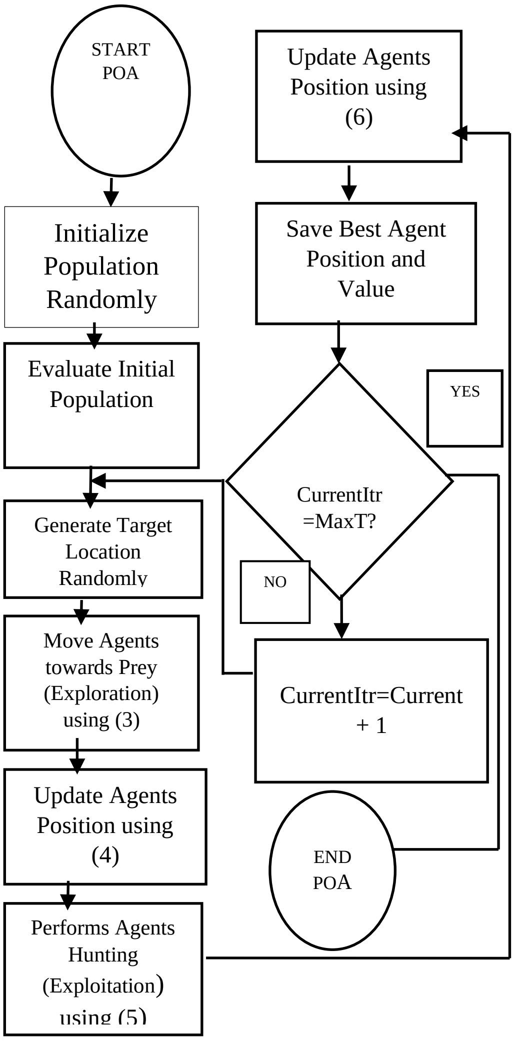

Figure 6: Flow diagram of optimized CNN for COVID-19 and pneumonia diagnosis.

EVALUATION METRICS.

Accuracy, recall, precision, specificity, and F1score are used to assess the efficacy of lung disease detection and classification algorithms.The Accuracy shows the proportion of successfully predicted samples to all samples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

The ratio of correctly classified positive samples to the total number of actually positive samples is known as the recall.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \quad (8)$$ Type equation here.

The percentage of accurately identified negative samples to the total number of genuine negative samples is known as specificity.

$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative} + \text{False Negative}} \quad (9)$$

The precision indicates the proportion of samples with a positive prediction that is positive.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (10)$$

The weighted harmonic mean of recall and precision is known as the F1-score

$$\text{F1 score} = 2\ \text{X}\ \frac{\text{Precision X Recall}}{\text{Precision} + \text{Recall}} \quad (11)$$

RESULTS AND DISCUSSIONS

The research utilized Tkinter framework tools in Python 3.9 to implement techniques for detecting and classifying COVID-19 and pneumonia. The system specifications included an Intel Core i5-CPU running at 2.60GHz, 6GB of RAM, a 500GB hard disk, and a 70KGPU graphics card. The operating system used was Windows 10 64-bit. A test dataset of 889 chest X-ray images was utilized to evaluate the approaches. This test dataset includes 452 cases of COVID-19 and 437 cases of pneumonia. An evaluation was conducted to compare the performance of a CNN with hyperparameter adjustment to that of the state-of-the-art CNN approach. Table 1 displays the contingency table that categorizes chest X-ray images using CNN with hyperparameter tuning and CNN without hyperparameter tuning. The data presented in Table 1 demonstrates that the CNN with hyperparameter tweaking had superior performance to the comparable false negatives. In the classification of the COVID-19 dataset, the CNN with hyperparameter tuning achieved superior performance by accurately recognizing 220 and 212 instances of the COVID-19 and pneumonia datasets respectively, compared to the 208 and 206 instances identified by the CNN without hyperparameter tuning. The CNN optimization resulted in a reduction in false positives. Table 2 displays CNN's performance metrics, including accuracy, recall, precision, specificity, and the F-1 score, without adjusting the hyperparameters. The result obtained in Table 1 reveals that the CNN with hyperparameter tuning technique achieved accuracy, recall, precision, specificity, and an F-1 score of 97.28%, 98.44%, 96.60%, 93.19%, and 95.09% for pneumonia and 97.00%, 98.26%, 98.68%, 94.21%, and 95.06% for COVID-19, respectively. Also, the CNN approach attained accuracy, recall, precision, specificity, and an F1-score of 90.04%, 91.04%, 89.31%, 90.71%, and 91.39% for pneumonia, and 91.94%, 91.94%, 92.16%, 91.42%, and 91.16%for COVID-19, respectively. It can be deduced from the results based on the performance metrics that the CNN with hyperparameter tuning outperformed the CNN model without hyperparameter tuning in the prediction of COVID-19 and pneumonia using chest X-rays. The results reveal that the CNN with hyperparameter adjustment obtains the best classification accuracy over the state-of-the-art CNN model. While CNN with hyperparameter tweaking achieves improved accuracy, it is evident that in terms of the particular disease, the ideal results are those with the lowest amount of false negatives and false positives.

A real-life interpretation of a false negative instance would result in the mistaken assumption that the patient is not infected with what this implications for the propagation of the virus and public health. False positives, on the other hand, would result in the false impression that the patient who is not infected is infected. It is obvious from the results obtained that the CNN with a hyperparameter tuning solution outperformed CNN without hyperparameter tuning and other state-of-the-art approaches (Zahedi *et al*., 2021, Rajagopal *et al*., 2023).

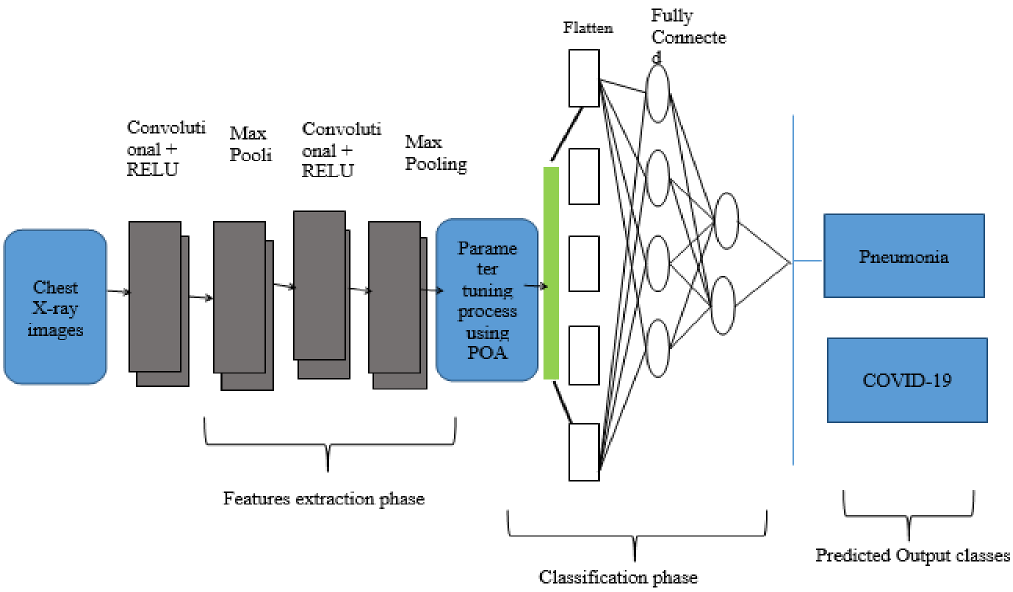

Figure 5. The proposed CNN-POA Model Architecture.

TABLE 1: Results of Trained CNN with hyperparameter tuning

True Positive	False Negative	False Positive	True Negative	Accuracy %	Recall %	Precision%	Specificity %	F1-Score%	Disease type
212	4	15	206	97.28	98.44	96.60	93.19	95.09	Pneumonia
220	7	13	212	97.00	98.26	98.68	94.21	95.06	Covid-19

TABLE 2: Trained CNN result without hyperparameter tuning

True Positive	False Negative	False Positive	True Negative	Accuracy %	Recall %	Precision%	Specificity %	F1-Score%	Disease type
206	22	20	202	90.04	91.04	89.31	90.71	91.39	Pneumonia
208	20	18	207	91.94	91.94	92.16	91.42	91.16	Covid-19

TABLE 3: Results comparison with related works

Authors	Model	Accuracy%	Recall%	Precision%	Specificity%	F1- Score%
Bharati *et al*., (2021)	CO-ResNet	90.90	-	90.20	-	-
Buvana *et al*., (2021)	BWO-CNN	95.43	95.8	-	94.10	95.60
Proposed work	POA-CNN	97.28	97.92	97.89	94.11	95.64

Also, it delivers a consistently high value of sensitivity, accuracy, and precision for patients with COVID-19 and pneumonia, both positive and negative. The fact that the suggested approach minimized false negatives and positives illustrates its robustness. Its performance could be improved by training on larger multi-geographical datasets. This deep learning tool may assist move the healthcare workflow toward deep learning methodologies. It was obvious that the performance of CNN with hyperparameter adjustment was well-matched to several other existing techniques, as indicated in Table 3. Hence, it will achieve a more accurate and computationally efficient approach, which will improve radiologists' interpretation and categorization of chest X-rays in diagnosing COVID-19 and pneumonia. The proposed technique is recommended for use in clinical practices for quick and accurate diagnosis of lung-related illnesses.

CONCLUSIONS

A technique to help radiologists automatically diagnose lung disorders such as pneumonia and COVID-19 is created in this research. Hyperparameter fine-tuning and data augmentation are applied despite the limited availability of data. Chest X-ray images were utilized to train CNN algorithms, with the POA-CNN model serving as the foundational model. The suggested method is more successful in diagnosing COVID-19 patients and pneumonia cases due to its increased accuracy rates for both pneumonia and COVID-19 disorders. An examination in comparison with previous research validated the efficacy of the suggested methodology. Other Deep Learning approaches and a hybrid metaheuristic optimization algorithm may be the subject of future research.

ACKNOWLEDGEMENTS

The authors would like to thank researchers and the National Institutes of Health (NIH) for uploading their datasets in Kaggle and github data repositories.

REFERENCE

Hashmi, F., Katiyar, S., Keskar, A. G., Bokde, N. D., and Geem, Z. W. (2020): Efficient Pneumonia detection in chest X-ray images using deep transfer learning. Diagnostics, vol. 10, no. 6, p. 417.

Jain, R., Nagrath, P., Kataria, G., SirishKaushik, V., and Hemanth, D. J. (2020): Pneumonia detection in chest X-ray images using Convolutional Neural Networks and transfer learning. Measurement, vol. 165, p. 108

Apostolopoulos, I. D., & Mpesiana, T. A. (2020). Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. Physical and Engineering Sciences in Medicine, 43(2), 635-640.

Rajagopal, R. K., Karthick, R., Meenalochini, P., & Kalaichelvi, T. (2023). Deep Convolutional Spiking Neural Network optimized with Arithmetic optimization algorithm for lung disease detection using chest X-ray images. Biomedical Signal Processing and Control, 79, 104197.

Bharati, S. Podder, P., Mondal, M. R., & Prasath, V. B. (2021): CO-ResNet: (2021). Optimized ResNet model COVID-19 diagnosis from X-ray images. Article. International Journal of Hybrid Intelligent Systems pp. 71-85 71 DOI: 10.3233/HIS-210008

Yang, L and Shami, A. (2020): Hyperparameter optimization of machine learning algorithms: Theory and practice. Neuro computing, vol. 415, pp. 295–316, doi: 10.1016/j.neucom.2020.07.061.

Shekar, B.H., and Dagnew, G., (2019): Grid Search-Based Hyperparameter Tuning and Classification of Microarray Cancer Data. Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), pp. 1–8, doi: 10.1109/ICACCP.2019.8882943.

Zahedi, L., Mohammadi, F. G., Rezapour, S., Ohland, M. W and Amini, M. H.(2021): Search Algorithms for Automated Hyper-Parameter Tuning., pp. 1–10, [Online]. Available: http://arxiv.org/abs/2104.14677.Alibrahim, H., and Ludwig, S. A. (2021): Hyperparameter Optimization: Comparing Genetic Algorithm against Grid Search and Bayesian Optimization. IEEE Congress on Evolutionary Computation (CEC), pp.1551–1559, doi: 10.1109/CEC45853.2021.9504761.

Trojovsky, P., Dehghani, M. (2022): Pelican Optimization Algorithm: A Novel Nature Inspired Algorithm for Engineering Applications. Sensors, Vol. 22, ID: 855, pp. 1-34.

Salih, O., Duffy, K.J. (2023): Optimization Convolutional Neural Network for Automatic Skin Lesion Diagnosis Using a Genetic Algorithm. Appl. Sci. 2023, 13, 3248. https://doi.org/10.3390/app13053248

Kaggle's Chest X-ray Images (Pneumonia) dataset 2020. [Online]. Available: https://www.kaggle.com/paultimothymooney/chest-X-ray-Pneumonia.