Natural Resources for Human Health
eISSN: 2583-1194

Original Research

Received 18 March 2026
Revised 25 April 2026
Accepted 24 May 2026

Natural Resources for Human Health 2026, Vol. 6 (3): 1021–1049

KEYWORDS:
Lung cancer classification; Computed tomography; Convolutional neural network; Hybrid optimization; Particle swarm optimization–Genetic algorithm–Simulated annealing; CNN–KNN classifier.

A Hybrid PSO–GA–SA Optimized CNN–KNN Framework for Automated Multiclass Lung Cancer Classification from CT Images

Rucha M. Alandikar[1*], Bhagwan D. Phulpagar[2], Rajesh D. Bharati[3]

[1*]Department of Computer Engineering, P.E.S. Modern College of Engineering, Pune, India. ruchaalandikar37@gmail.com
[2] Department of Computer Engineering, P.E.S. Modern College of Engineering, Pune, India. phulpagarbd@gmail.com
[3] Department of Computer Science, Dr. D.Y. Patil Institute of Technology, Pimpri, Pune, India. rdbharati@gmail.com

ABSTRACT: Lung cancer is still one of the biggest causes of cancer deaths globally and it is essential to diagnose it accurately and timely from computed tomography (CT) images to enhance patient survival. While convolutional neural networks have been found to have good potential in automated CT-based classification, the performance of the networks is very much dependent on hyperparameter tuning, and the conventional hyperparameter tuning methods, such as manual search, grid search, and random search are computationally expensive and inefficient in high-dimensional space. In this study, a hybrid Particle Swarm Optimization–Genetic Algorithm–Simulated Annealing (PSO–GA–SA) methodology has been proposed to optimize CNN hyperparameters and CNN–KNN was designed to perform the automated multiclass lung cancer classification. The comprehensive preprocessing of the CT images is conducted to enhance the discriminative feature representation, such as resizing, grayscale, Gaussian filtering, Otsu thresholding, Canny edge detection, Sobel edge enhancement and channel fusion. The hybrid is a combination of PSO's global exploration capabilities, GA's evolutionary refinement capabilities and SA's local fine-tuning capabilities to achieve a balance between exploration and exploitation in optimization processes.

These optimized parameters were tested on CNN, MobileNetV2, DenseNet121, InceptionV3 and Hybrid CNN-KNN using Augmented IQ-OTHNCCD where accuracy, precision, recall, F1-score, confusion matrix and learning curve were used to assess their performance. The hybrid PSO–GA–SA optimized CNN–KNN model exhibited the best performance with 100% accuracy, 99% precision, 100% recall, and 98% F1-score, outperforming the individually optimized and stand-alone models, indicating that it has promising potential as a reliable clinical decision-support tool.

1. INTRODUCTION

Lung cancer is one of the most common and deadly diseases in the world, and causes a large number of cancer deaths every year. Early diagnosis and prompt treatment is critical for patients' survival. One of the most important and useful modalities for detecting pulmonary abnormalities is computed tomography (CT) imaging, which is capable of displaying small nodules that frequently are not visible on conventional chest radiographs and is high-resolution. However, manually analysing CT images is a very difficult and time-consuming process, and is highly dependent on the expertise of the radiologist [1]. With the growing amount of medical imaging data, accurate diagnosis becomes more complex with delayed clinical decisions and risk of human error. Therefore, automated computer-aided diagnosis (CAD) systems using artificial intelligence (AI) have attracted a lot of attention for increasing diagnostic accuracy and decreasing healthcare professionals' workload [3].

The recent progress in deep learning has revolutionized medical image analysis with its capabilities of automated feature extraction and classification from imaging data. Convolutional Neural Networks (CNNs) have shown an outstanding ability to learn the hierarchical representations from the CT images without hand-crafted feature engineering [3]. Moreover, transfer learning techniques have successfully employed advanced deep learning models, like MobileNetV2, DenseNet121 and InceptionV3, to perform well in different medical image classification applications. The strength of the prediction though, is very sensitive to the choice of the suitable hyperparameters like number of convolutional filters, dense layer neurons, dropout rate, learning rate, and optimization strategy. Inappropriate parameter selections to these models can result in slow convergence, over fitting, poor generalization and higher computational costs, which hinders the use of deep learning models in clinical settings [4]. Hyperparameter optimization has thus become an important feature to optimize performance of deep learning frameworks. Standard optimization methods, such as manual tuning, grid search and random search, are high computation cost and are inefficient with dimensionality increase. The Metaheuristic Optimization Algorithms can be used as an effective alternative, as they are able to explore the large search space with relatively low computational complexity. One efficient optimization algorithm for exploring the entire space is the Particle Swarm Optimization (PSO), which is based on the collective and individual experiences of the particle swarm [5]. To find better solutions while maintaining diversity among members of the population, the Genetic Algorithm (GA) incorporates evolutionary mechanisms like selection, crossover and mutation. Simulated Annealing (SA) uses a probabilistic acceptance strategy which means that occasionally solutions that are not as good are accepted, which decreases the chance of convergence to local optima. While each optimization algorithm has its own set of strengths, none of them is without its drawbacks for optimization when used one at a time; these may include tendency to converge too soon, slow convergence speed or sensitivity to parameter initialization [6].

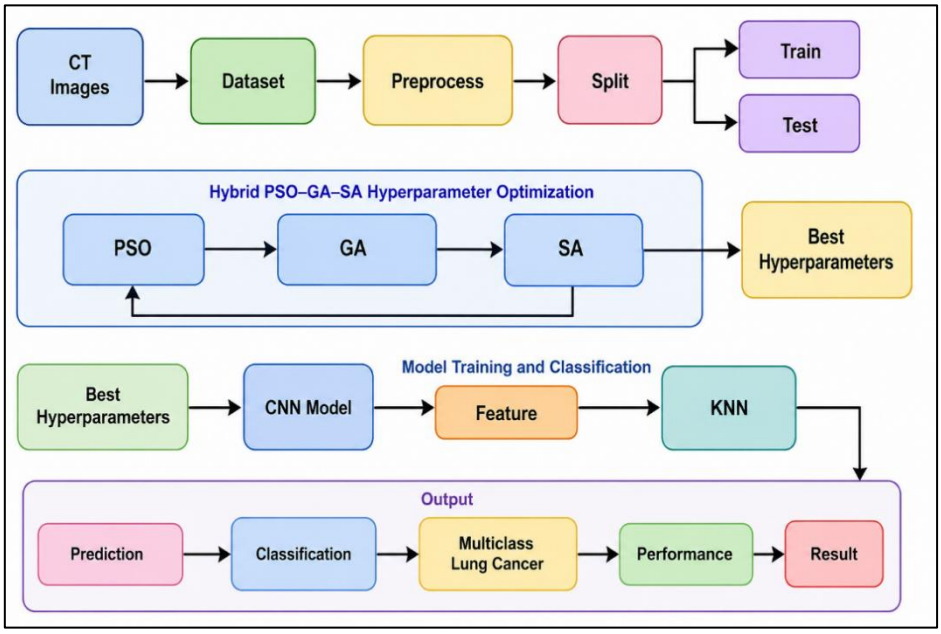

Figure 1. Workflow of the Proposed Hybrid PSO–GA–SA Hyperparameter Optimization

Figure 1 shows the proposed hybrid optimization process for the accurate multiclass lung cancer classification. In order to overcome these challenges, the present study introduces a hybrid PSO–GA–SA hyper-parameter optimization framework for automated multiclass lung cancer classification task using CT images, which is combined with a CNN based KNN classifier. In the proposed approach, the image preprocessing techniques are very comprehensive which include resizing the image, converting it to grey level, applying Gaussian filter, Otsu thresholding, Canny edge detection [7] and Sobel edge enhancement to enhance the images and to represent the features before the classification. The hybrid optimization approach takes advantage of PSO's global exploration ability, GA's evolutionary optimization ability and SA's local fine-tuning ability to obtain the best balance between exploration and exploitation for hyperparameter optimization. The optimized parameters are then used for training CNN, MobileNetV2, DenseNet121, InceptionV3 and hybrid CNN–KNN classifiers with an 80:20 ratio of training–testing data, respectively [8]. The performance evaluation is done by comparing the result with the other using confusion matrix, classification accuracy, validation loss and learning curve with individual and hybrid optimization strategies. The proposed

framework to achieve classification accuracy, convergence stability, model generalization and to reduce the manual tuning of hyperparameters provides computer-aided diagnostic solution for multiclass lung cancer classification from CT image which is robust, scalable and efficient [9].

2. RELATED WORK

With the fast development of AI and deep learning technology, automated lung cancer classification from CT image has been gain a lot of attention. Hand crafted texture, shape and intensity features were mostly used with machine learning classifiers. These methods have been moderately successful for classification, but the complexity of feature engineering and the poor adaptability to medical imaging datasets with mixed modalities limited the effectiveness of these methods [10, 11]. In recent years, the emergence of the powerful (CNNs) has mitigated these drawbacks by automatically learning the hierarchical feature representations directly from the CT images, thus achieving significant enhancement of accuracy and robustness in the task of classification. Various deep learning (DL) architectures have been studied for the purposes of lung cancer detection and classification. MobileNetV2 has proven to be more efficient in terms of computation and has less complex models, suitable for resource-limited environments [12]. The DenseNet121 model features dense connections to accomplish the advantages of feature reuse, which include effective propagation of gradients, and improve classification outcomes. A multi-scale convolutional network is used to extract discriminative image features from different receptive fields in the manner of multi-scale convolutional operations in InceptionV3, which helps to improve the multiclass recognition [13, 14]. Although these architectures proved to be promising in their performance, they are still heavily dependent on their hyperparameters, with convolutional filter size, learning rate, dropout ratio and network depth being among the main ones. When selecting these parameters, there is a tendency to over-fit, to have unstable convergence, and to not be as diagnostic as it could be [15].

In order to alleviate these challenges, many optimization algorithms have been embedded into the deep learning frameworks. Due to its good global search ability and rapid convergence, (PSO) has been widely used to optimize the hyper-parameters of CNN. The (GA) is an improvement on the optimization of the parameters by considering evolutionary operations like selection, crossover and mutation which provides better population diversity and exploration. Further optimization may be carried out using Simulated Annealing (SA) which probabilistically accepts the suboptimal solutions in the optimization process, easing the danger of being caught in local optimum solutions [16, 17]. These algorithms individually achieve the better performance of the models, but each has inherent weakness such as early convergence in PSO, slow evolutionary convergence in GA and sensitivity to the temperature in SA. Recently research has concentrated on hybrid optimization methods, using complementary metaheuristic optimization methods to get the best balance between global exploration and local exploitation. Most of the previous work, however, only combines two optimization techniques and rarely considers an extensive hybrid combination between multiple optimization techniques for multiclass lung cancer classification [18, 19]. In addition, there has been not much research on the fusion of optimized CNN feature learning and KNN classification in order to improve the diagnosis performance. Thus, the present study aims to introduce a hybrid PSO-GA-SA-based hyper parameter optimization system combined with CNN-KNN-based classifier into the system. The proposed method utilizes the PSO algorithm for global exploration, GA algorithm for evolutionary refinement and SA algorithm for the fine-tuning of the solution locally to enhance the stabilization of the convergence, selection of hyperparameters, classification accuracy and generalization of the model for automated multiclass lung cancer classification from CT images [20].

3. METHODOLOGY

The methodology proposed is a combination of image preprocessing, hybrid hyperparameter optimization, and deep learning-driven classification in order to achieve the automated and multiclass lung cancer detection from the CT images. Images are resized, converted to grayscale, Gaussian filtered and then thresholded using Otsu algorithm, enhanced using Canny edge and Sobel and finally channel fused. A hybrid PSO–GA–SA algorithm is used for optimization of CNN hyperparameters and a CNN–KNN framework is used for deep features extracted by CNN for classification. Performance is measured using comparative analysis, confusion matrix, accuracy and loss.

A. Load Dataset

The proposed framework will use the dataset for lung cancer from the Augmented IQ-OTHNCCD from the Kaggle repository. The data set consists of CT scan images of various lung diseases such as normal, benign and cancerous, which can be classified as multiclass classification [21]. When training, augmented samples can help to make the class balanced, mitigate overfitting,

and enhance the robustness of the model. The dataset is first split into training and testing set prior to the pre-processing and feature optimization steps. Figure 2 shows representative images of computed tomography (CT) scans of lungs with cancer in each of the diagnostic categories that were used to assess the diversity of the dataset. This variety of CT images can serve as a dependable reference for the performance of the proposed hybrid PSO–GA–SA optimized CNN–KNN framework [22] in terms of its effectiveness, generalization ability, and classification performance.

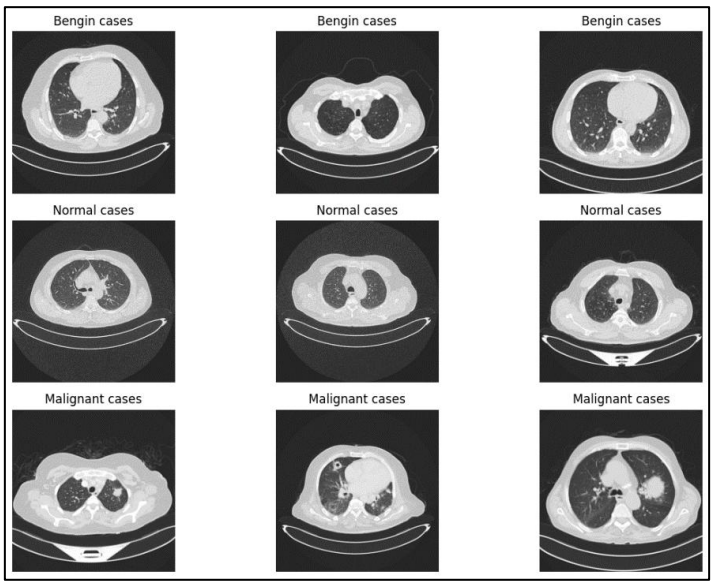

Figure 2. Lung Cancer Dataset Image Sample

B. Data Preprocessing

Data Preprocessing is used to preprocess the CT images for reliable deep learning, which includes image resizing, converting the image to a specific color space, grayscale transformation, Gaussian filtering, Otsu thresholding, Canny edge detection, Sobel enhancement, and channel merging. These are used to remove noise, enhance structural characteristics and provide standardized inputs which increases the accuracy of feature extraction and classification.

1. Image Resizing (128 * 128)

All CT scan images are resized to a common spatial resolution of 128 × 128 pixels, thus providing the deep learning models with equal dimension input. Image resizing not only saves the computational complexity but also saves the memory usage, and speeds up the network training without losing the key anatomical structures which are important for lung cancer classification. A fixed image size also makes it compatible with CNN-based architectures and allows to do batch processing when training and testing. This preprocessing step sets up a uniform representation of the input, which leads to a more efficient learning, a better convergence stability and better overall classification performance.

2. BGR to RGB Color Conversion

Before using the original CT images for further processing, the images are first converted from the (BGR) color format to the standard (RGB) format. This conversion will make sure that the images are interpreted correctly, because the image processing library (OpenCV) normally uses BGR order while most deep learning frameworks use RGB order. The correct order of color channels ensures that the information on the intensity of the image is properly preserved, does not distort features during pre-processing and remains unchanged throughout the various stages of the image processing. Therefore, RGB conversion is helpful to robustly extract the features and to facilitate the effective learning for convolutional neural network architectures.

3. Grayscale Color Conversion (RGB to GRAY)

After conversion to RGB, each CT image is converted to a monochromatic image with an intensity channel by converting the RGB channels. Because lung CT images contain structural and intensity information, but little colour information, the grayscale conversion eliminates redundant colour information, whilst preserving important anatomical information. This transformation

not only decreases the computational burden, but also eases the feature extraction process and increases the ability to obtain edges and textures in the following pre-processing steps such as Gaussian filtering, Otsu thresholding, Canny's edge detection and Sobel enhancement, which in turn would improve the robustness of lung cancer classification.

C. Image Enhancement and Feature Extraction

The image Enhancement and Feature Extraction is used to enhance the quality of the CT image using Gaussian filter, Otsu thresholding, Canny edge detection, Sobel gradient enhancement and channel fusion. They can remove noise, highlight the boundary of the lesions, maintain structural information and extract the discriminative features, which can be helpful in learning a robust representation and enhancing the classification accuracy in lung cancer.

1. Gaussian Blur

The CT images are first converted to grey images to reduce random noises and smooth the variations in intensity by applying Gaussian Blur in advance of feature extraction. The Gaussian filtering operation is a weighted averaging performed with a Gaussian kernel which will suppress high frequency components and preserve important anatomical structures. This pre-processing step reduces the impact of imaging artifacts on the creation of false edge responses and improves the ability of the subsequent segmentation approach to produce accurate segmentations. Gaussian Blur generates cleaner and more homogenous images resulting in more accurate thresholding, better localization of edges, more consistent features and final robustness of the proposed lung cancer classification framework.

2. OTSU Thresholding

To automatically segment the lung regions, the Otsu Thresholding method which aims to select an optimal global threshold by minimizing the intra-class intensity variance and maximizing the intensity variance between the classes is used. This adaptive thresholding method is very effective to separate the foreground structures from the background, without needing to set a threshold by hand. The outputs of this binary image are useful for identifying possible lung tissues with suspicious areas that can be more accurately extracted. In addition, the Otsu Thresholding technique helps to reduce the irrelevant background information, the quality of the image segmentation and gives a good basis for edge detection and classification in the proposed framework.

3. Canny Edge Detector

In order to detect the exact structure boundaries in lung CT images, the Canny Edge Detector is used in a multistage edge detection process. Gaussian smoothing and computation of gradient, non maximum suppression, double thresholding and edge tracking by hysteresis are applied to reduce noise first, followed by several steps of gradient computation and non maximum suppression, double thresholding and edge tracking by hysteresis. This technique preserves the major edges with a high fidelity and eliminates weak and noisy responses. The extracted edge information highlights lesion boundaries and tissue structures, which gives discriminating spatial features and boosts the lung cancer classification performance of a multiclass classification system when dealing with multiple classes of lung cancer.

4. Sobel Operator

The Sobel Operator is used to calculate the gradients of intensity in the image in both the horizontal and vertical directions, thus highlighting variations and orientations of structures in lung CT images. The operator highlights the regions with significant intensity change that represent anatomical boundaries and/or abnormal tissues based on the use of convolution kernels. Complementary texture and edge information is provided by the gradient images generated to increase feature discrimination. Therefore, the Sobel feature extraction is used to enhance the structural representation, which, in turn, helps in establishing robust learning by deep neural networks and enhances the accuracy of multiclass lung cancer classification.

5. Merge Channels (Threshold, Canny, Sobel)

The last stage of the pre-processing is the combination of the outputs of Otsu thresholding and Canny edge detection and Sobel gradient computation into a single multi-channel representation of images. In this channel fusion, complementary segmentation, boundary and gradient information are combined in a feature rich image, allowing the deep learning model to take advantage of multiple visual features at once. Merged representation provides better context understanding of lung tissues and morphology of lesions and maintains the discriminate structure details. This results in improved feature quality,

classification capability and generalization capability of the proposed hybrid PSO–GA–SA optimized CNN–KNN framework. Using the CT images, figure 3 shows the application of the preprocessing stages that improves the quality of the images for the purpose of lung cancer classification.

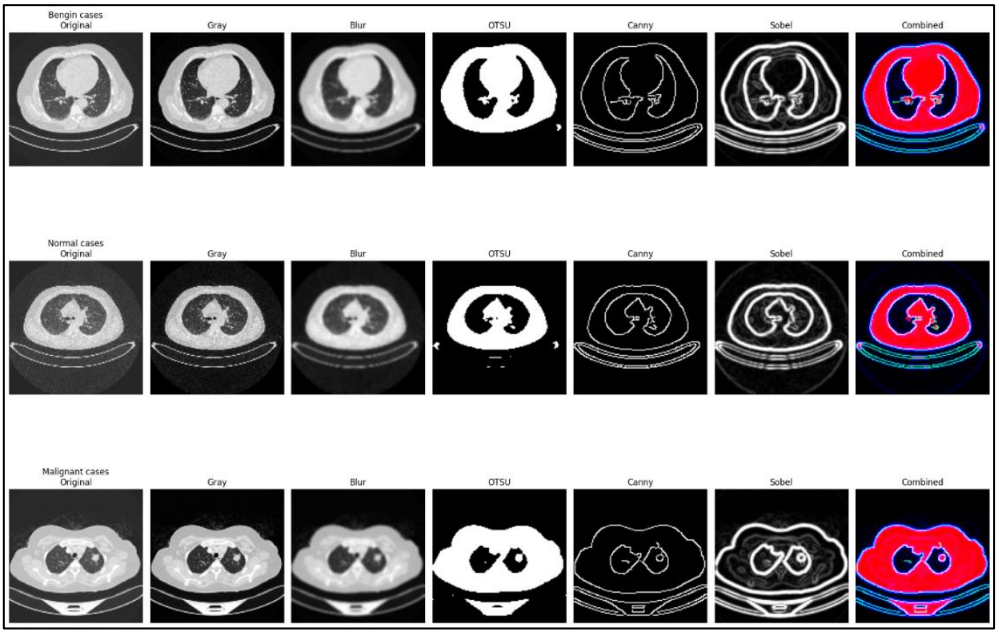

Figure 3. Pre-process Dataset Sample

D. Feature Optimization

For CNN hyperparameter optimization, Feature Optimization method is used with the following algorithms: PSO, GA, SA, and PSO–GA–SA hybrid. The optimization process not only reduces the loss during validation, but also improves the convergence, representation of features, accuracy of classification, robustness and generalization ability of the proposed framework.

1. Particle Swarm Optimization(PSO)

To optimize CNN hyperparameters, each particle in PSO is considered to be a candidate solution, which is defined in the following way: Convolutional filters, dense units, dropout rate, and learning rate. The position and velocity of each particle are randomly determined within the given search range and then adjusted using the individual best (pbest) solution and the global best (gbest) solution in terms of the validation loss. PSO iteratively exploits and explores to find the optimal combinations of hyperparameters, which leads to the rapid convergence, reduced classification error and improved predictive accuracy of the CNN–KNN framework.

- Algorithm: PSO-based Hyperparameter Optimization

Step	Description
1	Initialize swarm of particles (size = swarmsize), each particle represents CNN hyperparameters: [filters1, filters2, filters3, dense_units, dropout_rate, learning_rate] within given bounds
2	Randomly initialize particle positions within lower bounds (lb) and upper bounds (ub)
3	Initialize particle velocities randomly
4	Evaluate initial fitness for each particle using fitness_function(params)
5	Fitness is computed as validation loss after training CNN for few epochs

6	Set personal best (pbest) = current position of each particle
7	Set global best (gbest) = particle with minimum validation loss
8	For each iteration iter in 1 → maxiter
9	→ For each particle i in swarm
10	→ → Update velocity using formula:
11	→ → v[i] = w * v[i] + c1 * r1 * (pbest[i] - x[i]) + c2 * r2 * (gbest - x[i])
12	→ → Update position: x[i] = x[i] + v[i]
13	→ → Clip position within bounds [lb, ub]
14	→ → Convert integer parameters: filters & dense_units → int()
15	→ → Build CNN model using updated parameters
16	→ → Train model for small epochs (e.g., 3 epochs)
17	→ → Evaluate validation loss (fitness value)
18	→ → If current fitness < pbest[i], update pbest[i]
19	→ → If current fitness < gbest, update gbest
20	→ End particle loop
21	→ Print best fitness (lowest validation loss) in current iteration
22	End iteration loop
23	Select best parameters from gbest
24	Extract optimized hyperparameters
25	Build final CNN model using best parameters
26	Train final model on full training data (with more epochs)
27	Evaluate final model on validation/test data
28	Output final accuracy, loss, confusion matrix, classification report

2. Genetic Algorithm(GA)

The GA is used to optimize CNN hyperparameters by mimicking the evolution process of natural selection. Individuals in the population are candidate sets of hyper-parameters corresponding to convolutional filters, dense units, dropout rate and learning rate. Validation loss is used as a fitness function to assess the candidate solution. The tournaments, the blend crossover and the Gaussian mutation are used repeatedly to produce better offspring, without reducing diversity in the population. The optimized solution provided by GA helps with the model convergence, accuracy of classification and generalization capability of the proposed CNN-KNN framework.

- Algorithm: GA-based Hyperparameter Optimization

Step	Description
1	Initialize population of individuals (size = population size), each individual represents CNN hyperparameters: [filters1, filters2, filters3, dense_units, dropout_rate, learning_rate]

2	Randomly generate individuals using defined ranges: filters1 [16–64], filters2 [32–128], filters3 [64–256], dense_units [64–256], dropout [0.2–0.6], lr [0.00001–0.001]
3	Evaluate fitness of each individual using evaluate(individual)
4	Fitness = validation loss obtained after training CNN for few epochs (e.g., 3 epochs)
5	Store fitness as a tuple (since GA minimizes loss)
6	For each generation gen in 1 → ngen
7	→ Select parent individuals using Tournament Selection (tournsize = 3)
8	→ Apply crossover with probability cxpb = 0.5 using Blend Crossover (cxBlend)
9	→ Apply mutation with probability mutpb = 0.2 using Gaussian Mutation (mutGaussian)
10	→ Ensure mutated values remain within defined bounds
11	→ Convert integer parameters (filters, dense_units) using int()
12	→ Generate new offspring population
13	→ Evaluate fitness of new individuals (train CNN for small epochs)
14	→ Replace old population with new population
15	→ Track best individual (minimum validation loss) in current generation
16	End generation loop
17	Select best individual using selBest(population, k=1)
18	Extract optimized hyperparameters from best individual
19	Build final CNN model using best parameters
20	Train final model on full dataset with higher epochs
21	Evaluate final model using validation/test data
22	Output performance metrics: accuracy, loss, confusion matrix, classification report

3. Simulating Annealing(SA)

The SA is used as a local search optimization method to optimize CNN hyperparameters to minimize CNN validation loss. The algorithm starts with an initial solution, and then produces neighbouring solutions by making small changes to the parameters. Good solutions are accepted directly and bad ones are accepted probabilistically using a temperature-dependent acceptance function to break out of local optima. Temperature slowly decreases during the optimization process, leading to better stability of the solutions. Thereby, SA optimizes hyperparameters, boosts the convergence speed and the accuracy of classification of the CNN–KNN framework.

- Algorithm: SA-based Hyperparameter Optimization

Step	Description
1	Initialize a random solution representing CNN hyperparameters: [filters1, filters2, filters3, dense_units, dropout_rate, learning_rate] within given ranges
2	Evaluate initial solution using evaluate_solution(solution)
3	Compute fitness = validation loss after training CNN for few epochs (e.g., 3 epochs)

4	Set current_solution = initial solution
5	Set best_solution = current_solution
6	Initialize temperature T = initial_temp
7	For each iteration i in 1 → max_iter
8	→ Generate neighbor solution using get_neighbor(current_solution)
9	→ Modify one random parameter within its valid range
10	→ Evaluate new solution (train CNN for small epochs)
11	→ Compute new fitness = new validation loss
12	→ If new_loss < current_loss → accept new solution
13	→ Else compute acceptance probability:
14	→ → prob = exp((current_loss - new_loss) / T)
15	→ → Generate random number r ∈ [0,1]
16	→ → If r < prob → accept worse solution (to escape local minima)
17	→ Update current_solution and current_loss if accepted
18	→ If current_loss < best_loss → update best_solution
19	→ Reduce temperature: T = T × cooling_rate
20	→ Print best loss for current iteration
21	End iteration loop
22	Return best_solution (optimized hyperparameters)
23	Extract best parameters (filters, dense_units, dropout, learning_rate)
24	Build final CNN model using optimized parameters
25	Train final model with full epochs on training data
26	Evaluate model on validation/test dataset
27	Output performance metrics: accuracy, loss, confusion matrix, classification report

4. PSO+GA+SA Hybrid

The proposed hybrid PSO–GA–SA optimization framework first applies the PSO and GA and finally SA to get the optimum hyperparameters of the CNN. PSO searches the whole solution space to find the most promising search space areas; GA is used for local tuning and close to optimum solutions using evolutionary operations, and finally, SA performs local tuning to avoid local optima and accelerate convergence. The fitness function used in the optimization is the validation loss. The optimal hyperparameters are then used to train the CNN–KNN classifier, which gives better classification accuracy, robustness, stability of convergence and generalization of the model.

- Algorithm: Hybrid PSO + GA + SA for Hyperparameter Optimization

Step	Description
1	Define solution vector representing CNN hyperparameters: [filters1, filters2, filters3, dense_units, dropout_rate, learning_rate]
2	Define parameter bounds and apply clip_solution() to keep values within valid ranges
3	Define fitness function evaluate_solution(solution) using validation loss from CNN training (few epochs)
Phase 1: Particle Swarm Optimization (PSO – Global Search)	
Step	Description
4	Initialize swarm of particles randomly within bounds
5	Initialize velocities of particles to zero
6	Evaluate fitness of each particle
7	Set personal best (pbest) = current particle positions
8	Set global best (gbest) = particle with minimum validation loss
9	For each iteration in PSO
10	→ For each particle i
11	→ → Update velocity:
12	→ → v[i] = w*v[i] + c1*r1*(pbest[i]-x[i]) + c2*r2*(gbest-x[i])
13	→ → Update position: x[i] = x[i] + v[i]
14	→ → Apply clipping to maintain bounds
15	→ → Evaluate new fitness
16	→ → Update pbest if improved
17	→ End particle loop
18	→ Update gbest
19	End PSO loop
20	Output pso_best solution
Phase 2: Genetic Algorithm (GA – Refinement)	
Step	Description
21	Initialize GA population around pso_best by adding small random noise
22	Evaluate fitness of all individuals
23	For each generation in GA

24	→ Select parents using Tournament Selection
25	→ Apply crossover (Blend Crossover) with probability cxpb
26	→ Apply mutation (Gaussian Mutation) with probability mutpb
27	→ Clip solutions to valid bounds
28	→ Evaluate new offspring fitness
29	→ Replace population with new individuals
30	End GA loop
31	Select best individual → ga_best
Phase 3: Simulated Annealing (SA – Fine-Tuning)	
Step	Description
32	Initialize current solution = ga_best
33	Set best solution = current solution
34	Initialize temperature T
35	For each iteration in SA
36	→ Generate neighbor solution by slightly modifying one parameter
37	→ Apply clipping to maintain bounds
38	→ Evaluate new fitness
39	→ If new_loss < current_loss → accept solution
40	→ Else compute probability: exp((current_loss - new_loss)/T)
41	→ Accept worse solution with this probability
42	→ Update best solution if improved
43	→ Reduce temperature: T = T × cooling_rate
44	End SA loop
45	Output final optimized solution → final_best
Final CNN Model Training	
Step	Description
46	Extract optimized hyperparameters from final_best
47	Build final CNN model using optimized parameters
48	Train model on full dataset with sufficient epochs
49	Evaluate model on validation/test data
50	Output final performance: accuracy, loss, confusion matrix, classification report

Natural Resources for Human Health
eISSN: 2583-1194

Table 1. Hyper Parameters of Optimisation Models

Parameters	PSO value	GA value	SA value	PSO+GA+SA value
Filters1	62	31	44	24
Filters2	68	43	91	76
Filters3	96	214	112	123
Dense_Units	166	188	204	81
Dropout_Rate	0.3	0.05	0.257	0.2
Lr	0.001	0.00029	0.0006	1e-05

The optimized hyperparameters by PSO, GA, SA and PSO–GA–SA are summarized in Table 1. The hybrid approach chooses 24, 76 and 123 filters, 81 dense units, 0.2 dropout and learning rate of 1×10^{-5} to achieve a trade-off between convergence and prevention of overfitting while simultaneously improving the overall lung cancer classification performance.

E. Train Test Split

In order to have a reliable model developed and a good performance evaluation, the preprocessed CT image dataset is separated into 80% training data and 20% testing data. The training set is used to obtain the parameters for the network and to find the optimum hyperparameters by the proposed PSO–GA–SA framework, while the testing set is not seen during the training and is only used for evaluating the performance of the network. This partitioning helps in getting enough training samples and also ensures independent evaluation set and the classification accuracy, generalization ability, robustness and predictive reliability of the proposed CNN–KNN classification framework can be measured accurately.

F. Classification Models

KNN, CNN, MobileNetV2, DenseNet121, InceptionV3 and hybrid CNN–KNN classifiers are evaluated for multiclass lung cancer detection in Classification Models. Comparative analysis shows that the optimized CNN–KNN model outperforms all the other models in terms of classification accuracy, precision, recall and F1-scores, exhibiting high predictive performance, robustness and generalization performance.

1. KNN

The KNN (K-Nearest Neighbors) classifier classifies a lung CT image based on the class with the highest number of feature vectors in the training set when compared to the feature vectors of the K-nearest neighboring CT images in the test set. The Euclidean distance is used to calculate similarity of samples, and a majority voting is used for the final class label. In the proposed lung cancer classification framework, KNN is able to classify high-level CNN features with low training complexity and offers strong decision boundaries, and enhance the multiclass discrimination.

$$d(x,y) = \sqrt{[\Sigma(i=1\rightarrow n)(xi-yi)^2]}$$

$$Nk(x) = argmin\ d(x,xi)$$

$$\hat{y} = mode\{yi: xi \in Nk(x)\}$$

2. CNN Model

The CNN [24] automatically obtains the hierarchical representation of images by means of convolution, nonlinear activation and pooling. To extract discriminative texture and structural information, multiple convolutional layers are used and then fully connected layers are employed to predict multiple classes.

$$F = X * K + b$$

$$A = ReLU(F) = max(0,F)$$

$$\hat{y} = Softmax(WA + b)$$

CNN well captures spatial features of both normal and both non-cancerous and cancerous lung tissues, which results in powerful deep features for the accurate classification of normal, non-cancerous and cancerous lung tissue, and then subsequently hybrid optimization by the proposed PSO–GA–SA framework. The CNN architecture which uses hierarchical feature extraction for accurate lung cancer classification is shown in figure 4.

Model: "sequential_31"

Layer (type)	Output Shape	Param #
conv2d_93 (Conv2D)	(None, 126, 126, 22)	616
max_pooling2d_93 (MaxPooling2D)	(None, 63, 63, 22)	0
conv2d_94 (Conv2D)	(None, 61, 61, 128)	25,472
max_pooling2d_94 (MaxPooling2D)	(None, 30, 30, 128)	0
conv2d_95 (Conv2D)	(None, 28, 28, 232)	267,496
max_pooling2d_95 (MaxPooling2D)	(None, 14, 14, 232)	0
flatten_31 (Flatten)	(None, 45472)	0
dense_62 (Dense)	(None, 256)	11,641,088
dropout_31 (Dropout)	(None, 256)	0
dense_63 (Dense)	(None, 3)	771

Total params: 11,935,443 (45.53 MB)
Trainable params: 11,935,443 (45.53 MB)
Non-trainable params: 0 (0.00 B)

Figure 4. CNN Model Architecture Summary

3. MobileNetV2 Model

In this paper, the lightweight deep convolutional network, MobileNetV2 [23] [25], is introduced, which uses inverted residual blocks and depthwise separable convolutions to achieve lightweight without losing the classification accuracy. Medical image analysis requires a network that can be used to efficiently represent the crucial feature mappings, thus linear bottleneck layers is introduced.

$$Y = D(X) + P(X)$$

$$P(X) = D(X) * Kp$$

These discriminative features extracted are used to classify lung cancer in the multiclass lung cancer classification, having high computational efficiency and generalization on CT image samples. The MobileNetV2 architecture for efficient feature learning with low computational complexity is shown in figure 5.

Model: "sequential"

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_128 (Functional)	(None, 4, 4, 1280)	2,257,984
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 1280)	0
dense_1 (Dense)	(None, 256)	327,936
dropout_1 (Dropout)	(None, 256)	0
dense_2 (Dense)	(None, 3)	771

Total params: 2,586,691 (9.87 MB)
Trainable params: 1,855,107 (7.08 MB)
Non-trainable params: 731,584 (2.79 MB)

Figure 5. MobileNetV2 Model Architecture Summary

4. DenseNet121 Model

DenseNet121 [26] improves feature propagation by making dense connections between adjacent convolution layers, allowing a layer to have access to the features learned by the previous layers. This architecture helps mitigate gradient vanishing, reuse features and cuts down on redundant learning with lower number of parameters.

$$xl = Hl([x0, x1, \ldots, xl - 1])$$

$$Hl(x) = Wl * x$$

$$\hat{y} = Softmax(Wxl + b)$$

The dense connectivity leads to better learning and representation from lung CT images and achieves superior diagnostic classification performance, fast convergence speed, and stable multiclass prediction performance in the proposed diagnostic framework. The DenseNet121 architecture for efficient feature reuse towards improved classification accuracy is shown in figure 6.

Model: "sequential_1"

Layer (type)	Output Shape	Param #
densenet121 (Functional)	(None, 4, 4, 1024)	7,037,504
global_average_pooling2d_2 (GlobalAveragePooling2D)	(None, 1024)	0
dense_3 (Dense)	(None, 256)	262,400
dropout_2 (Dropout)	(None, 256)	0
dense_4 (Dense)	(None, 3)	771

Total params: 7,300,675 (27.85 MB)
Trainable params: 904,579 (3.45 MB)
Non-trainable params: 6,396,096 (24.40 MB)

Figure 6. DenseNet121 Model Architecture Summary

5. InceptionV3 Model

As a part of the inception module, InceptionV3 [extracts features at multiple scales by using multiple convolution kernels at the same time. Factorized convolutions not only save computational resources but also maintain feature diversity; this has allowed to effectively capture fine and coarse anatomical structures in CT images.

$$F = [F1 \times 1, F3 \times 3, F5 \times 5, Fpool]$$

$$\hat{y} = Softmax(WFout + b)$$

The architecture learns rich multilevel representations which enhance the accuracy of multiclass lung cancer classification, computational efficiency and the ability of the network to generalize across heterogeneous imaging patterns. The architecture InceptionV3 is presented in Figure 7, which extracts the multiscale features from the images for robust lung cancer classification.

Model: "sequential_3"

Layer (type)	Output Shape	Param #
inception_v3 (Functional)	(None, 2, 2, 2048)	21,802,784
global_average_pooling2d_4 (GlobalAveragePooling2D)	(None, 2048)	0
dense_7 (Dense)	(None, 256)	524,544
dropout_4 (Dropout)	(None, 256)	0
dense_8 (Dense)	(None, 3)	771

Total params: 22,328,099 (85.17 MB)
Trainable params: 525,315 (2.00 MB)
Non-trainable params: 21,802,784 (83.17 MB)

Figure 7. InceptionV3 Model Architecture Summary

6. CNN + KNN Hybrid Model

The proposed CNN–KNN hybrid model incorporates the deep feature extraction and distance-based classification for achieving better multiclass lung cancer recognition. CNN learns discriminative feature representations automatically from CT images, while KNN takes the learned discriminative feature vectors for final classification.

$$f = CNN(X)$$

$$di = ||f - fi||2$$

$$\hat{y} = mode\{yi: di \in K\}$$

This is a hybrid approach that improves the separability of the features, decreases the ambiguity in the classification, makes it more robust to complex image variations and provides better predictive performance than from individual classification models.

4. RESULT ANALYSIS

Result Analysis is comparative evaluation of PSO, GA, SA and Hybrid PSO–GA–SA optimized classification models based on accuracy, precision, recall and f1 score. The experimental results show the superiority of the hybrid PSO–GA–SA optimized CNN–KNN framework over the single optimization approaches and stand-alone deep learning models in terms of the most accurate prediction, robustness, and generalization.

A. Confusion Matrix (No Optimisation Technique)

The confusion matrices for CNN, MobileNetV2, DenseNet121, InceptionV3, KNN and hybrid CNN–KNN models without hyperparameter optimization are shown in figure 8(a–f). Noticeable inter-class misclassifications are visible in the CNN, MobileNetV2, DenseNet121, and InceptionV3 models, especially among the benign and normal classes, which compromises the general usefulness of the models.

Natural Resources for Human Health
eISSN: 2583-1194

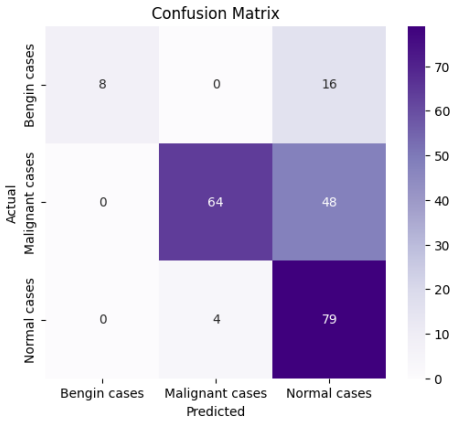

Fig 8(a). Confusion Matrix of CNN Model

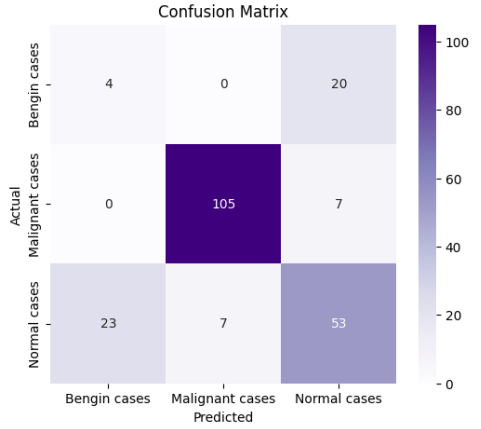

Fig 8(b). Confusion Matrix of MobileNetV2 Model

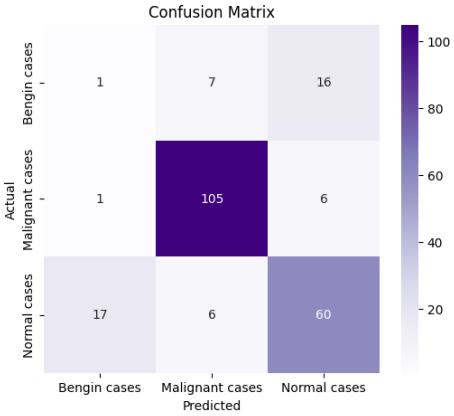

Fig. 8(c). Confusion Matrix of DenseNet121 Model

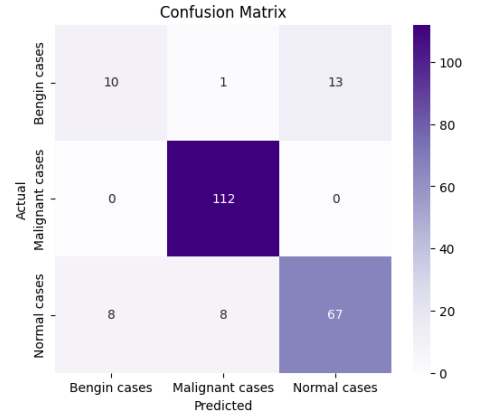

Fig. 8(d). Confusion Matrix of InceptionV3 Model

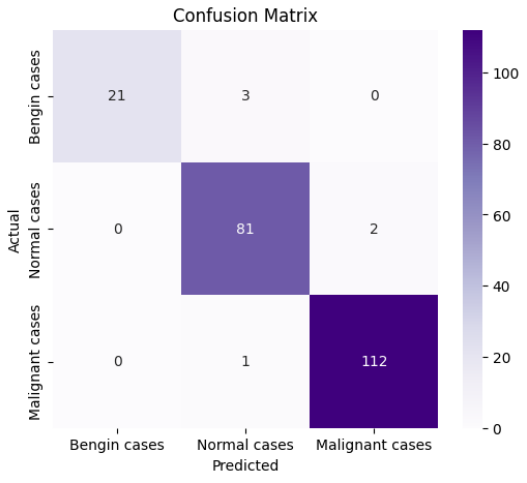

Fig. 8(e). Confusion Matrix of KNN Model

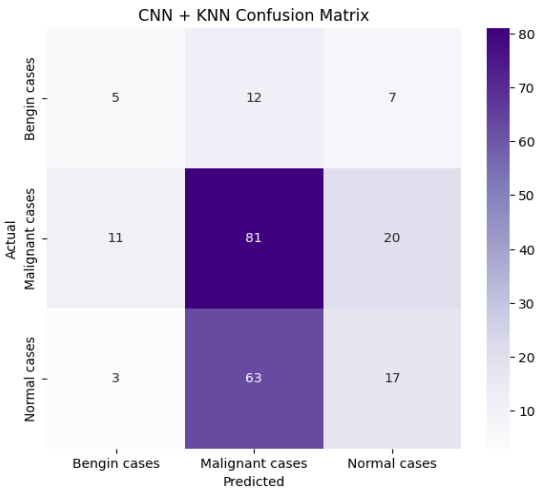

Fig. 8(f). Confusion Matrix of CNN + KNN Model

Natural Resources for Human Health
eISSN: 2583-1194

The standalone KNN model can be seen to provide better class separation but result in a few misclassifications. Before applying the optimization techniques, CNN–KNN hybrid model has the best performance with the lowest confusion matrix and the highest correct predictions with the least misclassified samples among all the models, which means this model has the best feature discrimination and multiclass classification ability.

B. Confusion Matrix (PSO Optimisation Technique)

The confusion matrices of PSO-optimized CNN–KNN, DenseNet121, InceptionV3 and MobileNetV2 models are shown in figures 9(a–e), respectively. When the hyperparameters are optimised using Particle Swarm Optimization, the class discrimination improves while misclassification decreases over non-optimised hyperparameters.

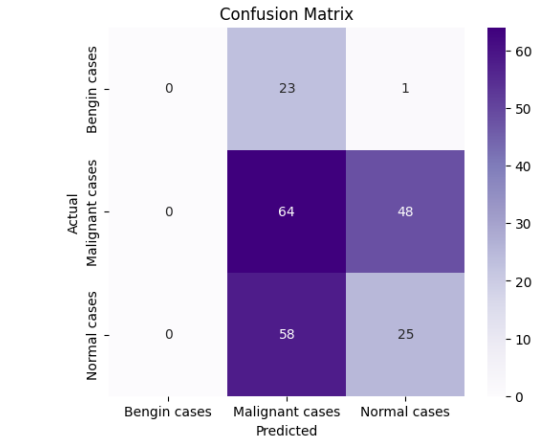

Figure 9(a). Confusion Matrix of MobieNetV2 + PSO

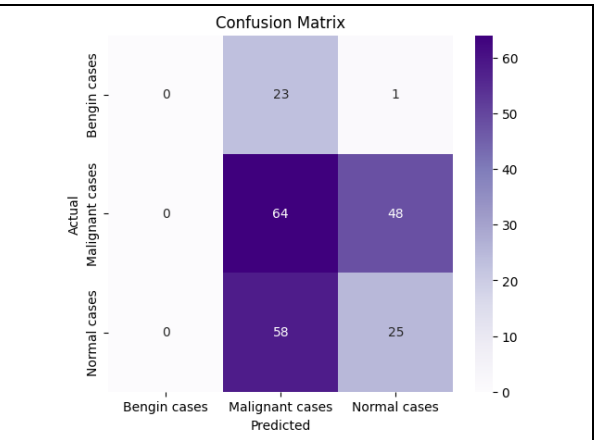

Figure 9(b). Confusion Matrix of MobieNetV2 + PSO

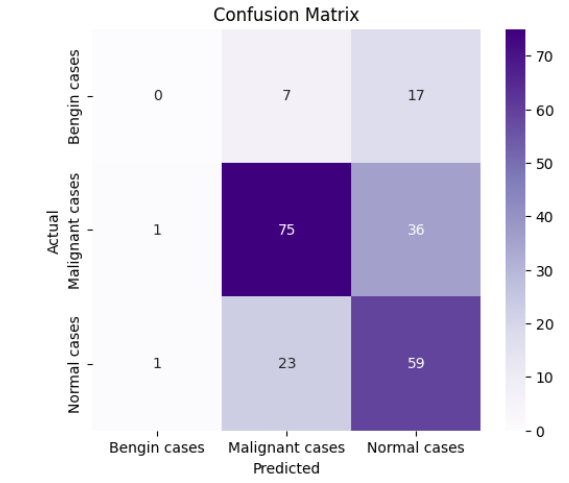

Figure 9(c). Confusion Matrix of DenseNet121 + PSO

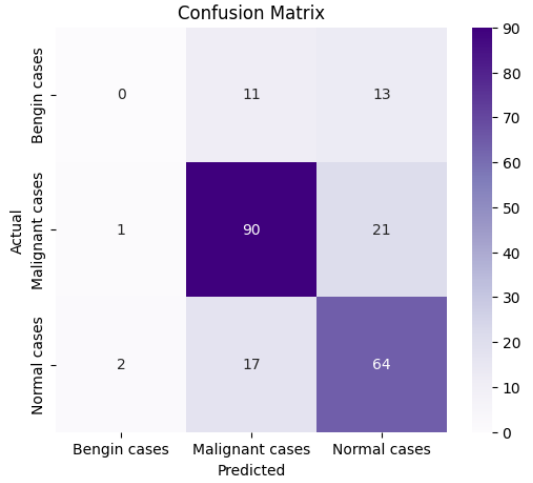

Figure 9(d). Confusion Matrix of InceptionV3 + PSO

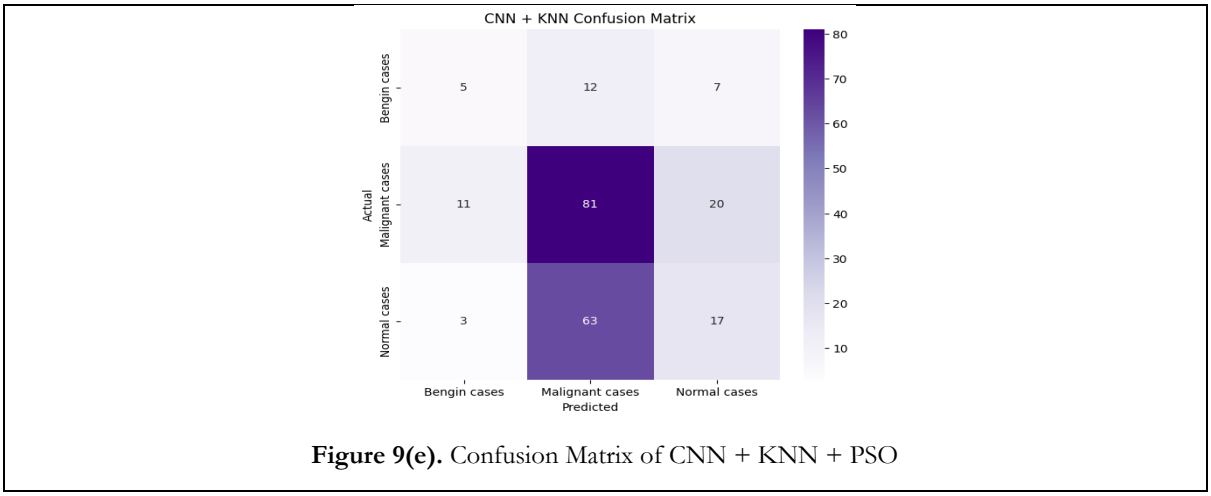

Figure 9(e). Confusion Matrix of CNN + KNN + PSO

The results of CNN, MobileNetV2, DenseNet121 and InceptionV3 show better recognition rate for both CT images of benign and malignant tumors and those of normal CT images, while there is a little inter-class confusion. The PSO optimized CNN–KNN model is the most accurate one and has the highest accuracy with the minimum amount of misclassifications, while also having the maximum number of correct predictions. All this shows the benefit of the PSO algorithm on the feature learning, convergence and multiclass lung cancer detection performance of the CNN–KNN model.

C. Confusion Matrix of GA

The confusion matrix of the GA optimized CNN, MobileNetV2, DenseNet121, InceptionV3 and CNN–KNN models are shown in figure 10(a–e). By using the Genetic Algorithm optimization, it is possible to achieve better consistency of the classification and reduce the confusion between the classes in all architectures, while identifying better hyperparameters.

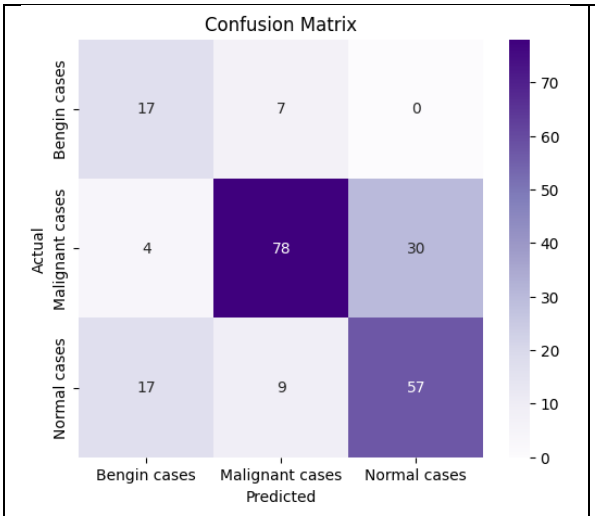

Figure 10(a). Confusion Matrix of CNN + GA

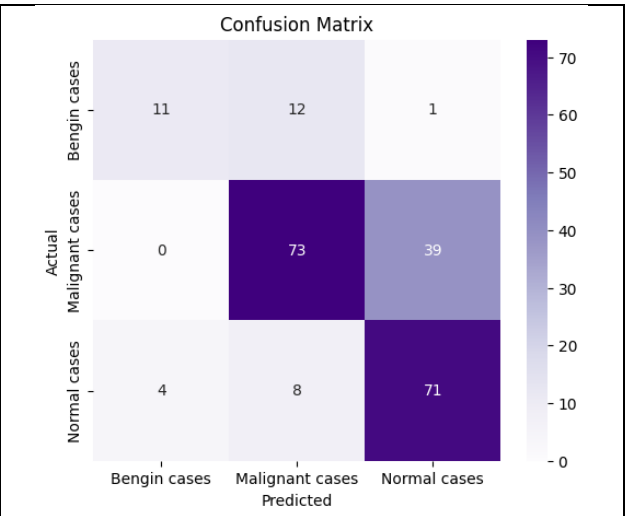

Figure 10(b). Confusion Matrix of MobileNetV2 + GA

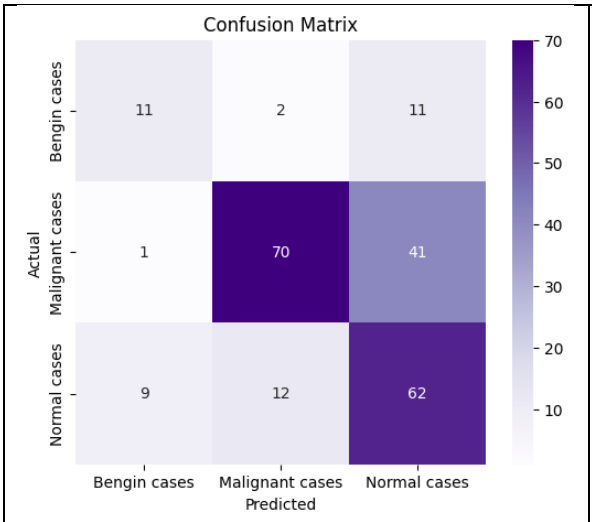

Figure 10(c). Confusion Matrix of DenseNet121 + GA

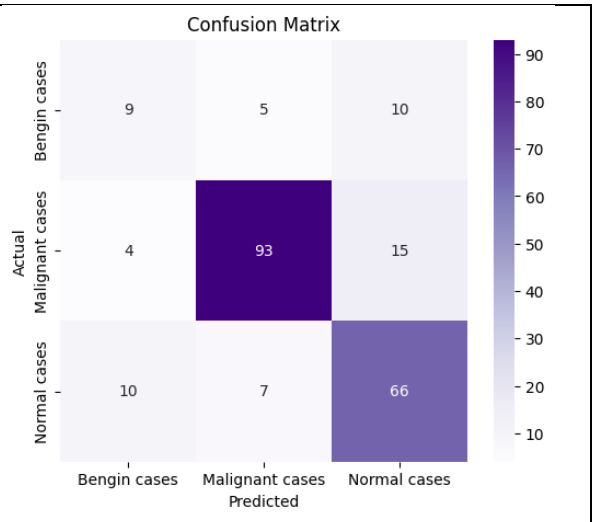

Figure 10(d). Confusion Matrix of InceptionV3 + GA

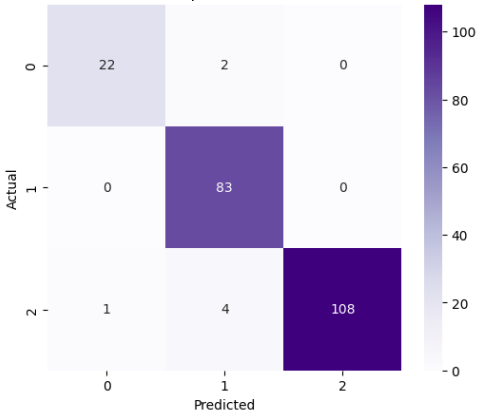

Figure 10(e). Confusion Matrix of CNN + KNN + GA

Compared with CNN and DenseNet121, MobileNetV2 and InceptionV3 have improved the recognition of benign, malignant and normal CT images. The GA-optimized CNN–KNN hybrid yields the best results in terms of the number of correctly classified samples while generating the least number of false predictions, which validates the fact that evolutionary optimization plays a significant role in enhancing the feature representation, convergence behavior and multiclass lung cancer classification performance.

D. Confusion Matrix of SA

The confusion matrices of the SA-optimized CNN, MobileNetV2, DenseNet121, InceptionV3 and CNN–KNN models are shown in Fig. 11(a–e), respectively. Simulated Annealing is used to refine hyperparameters locally and thus makes the classification more stable and the prediction errors are smaller in all architectures.

Natural Resources for Human Health
eISSN: 2583-1194

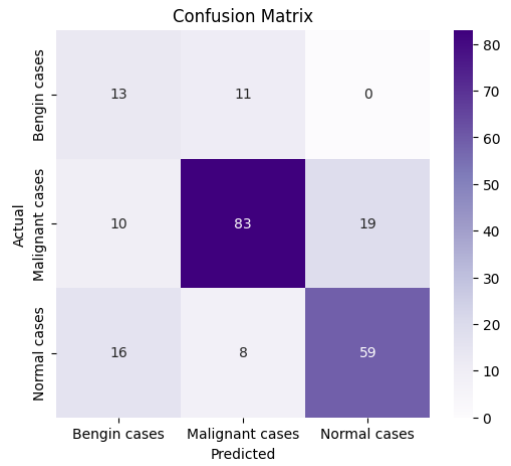

Figure 11(a). Confusion Matrix of CNN + SA

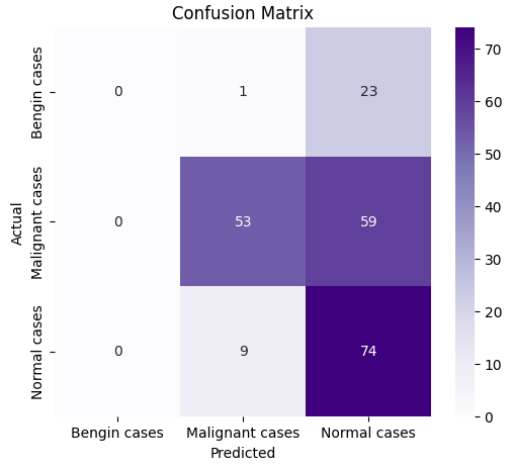

Figure 11(b). Confusion Matrix of MobileNetV2 + SA

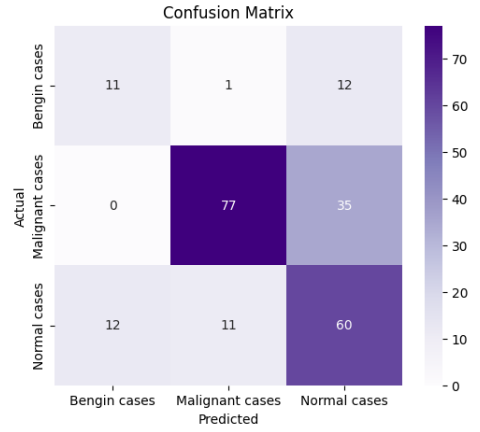

Figure 11(c). Confusion Matrix of DenseNet121 + SA

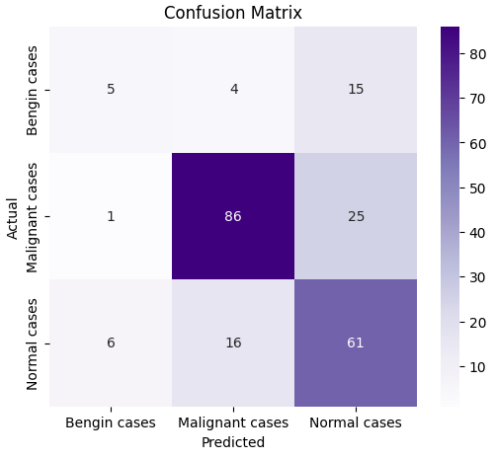

Figure 11(d). Confusion Matrix of InceptionV3 + SA

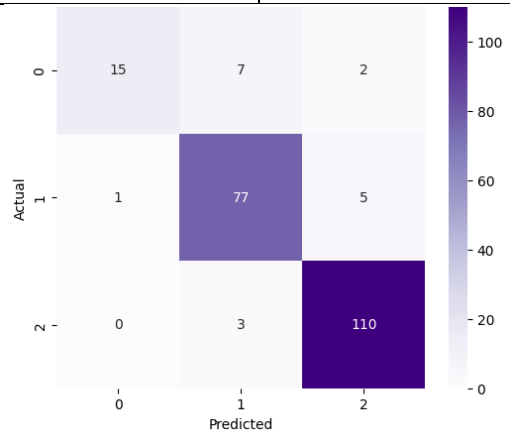

Figure 11(e). Confusion Matrix of CNN + KNN + SA

CNN, MobileNetV2, DenseNet121 and InceptionV3 still have moderate misrecognition of benign cases as normal, but the accuracy of the recognition is increased. The results of the SA-optimized CNN–KNN hybrid show the maximum number of correctly classified samples and minimum number of misclassifications, which proves that the performance of Simulated Annealing can effectively improve the quality of convergence, the ability of feature discrimination, and the accuracy of multiclass lung cancer classification.

E. Confusion Matrix of PSO + GA + SA

The confusion matrix of the hybrid PSO–GA–SA optimized CNN, MobileNetV2, DenseNet121, InceptionV3 and CNN–KNN are shown in figure 12(a–e). This hybrid optimization framework is able to include global exploration, evolutionary refinement and local fine tuning, significantly reducing inter-class misclassification in all the architectures.

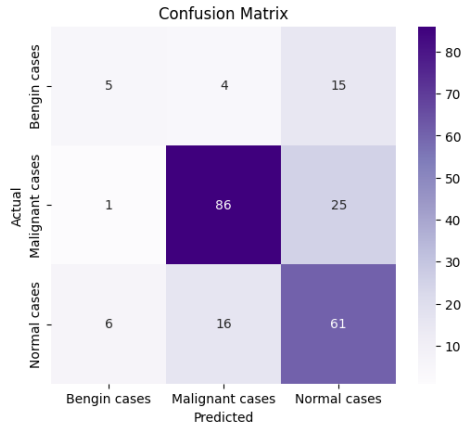

Figure 12(a). Confusion Matrix of CNN + PSO+ GA+ SA

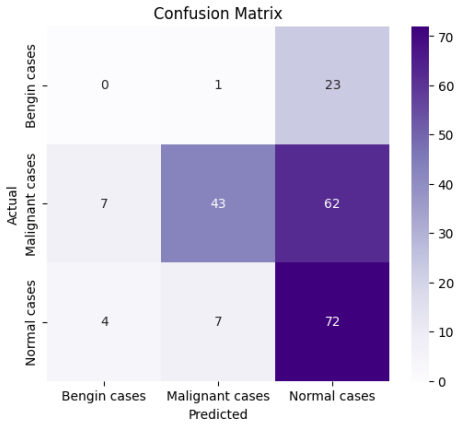

Figure 12(b). Confusion Matrix of MobileNetV2 + PSO+ GA+ SA

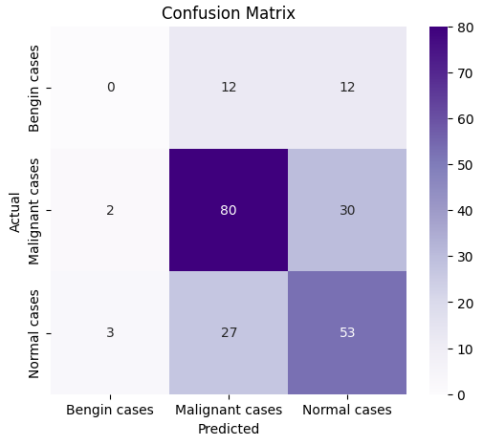

Figure 12(c). Confusion Matrix of DenseNet121 + PSO+ GA+ SA

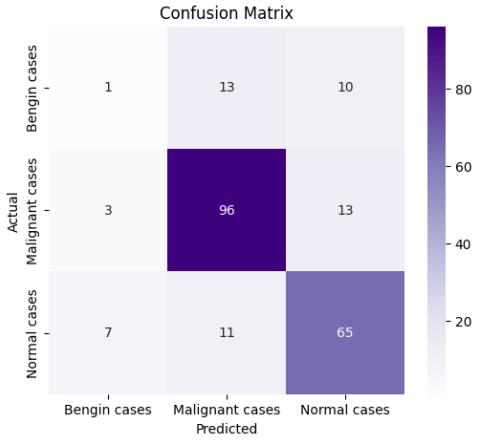

Figure 12(d). Confusion Matrix of InceptionV3 + PSO+ GA+ SA

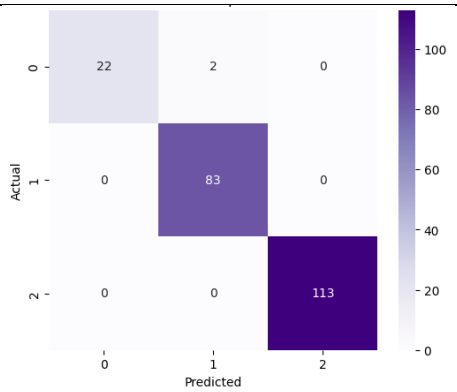

Figure 12(e). Confusion Matrix of CNN + KNN + PSO+ GA+ SA

CNN, MobileNetV2, DenseNet121, and InceptionV3 outperform the individual optimization approaches for the recognition of benign, malignant and normal CT images. The convergence of the hybrid CNN–KNN model is highest and the number of correct sample classifications is almost 100%, while error rate is almost negligible, which shows that the model has excellent convergence, feature representation and robustness, and excellent multiclass lung cancer classification performance.

F. Accuracy Loss Curve Graphs of PSO + GA + SA

The training and validation accuracy curves and the corresponding loss curves of the hybrid PSO–GA–SA optimized CNN model for 10 epochs is shown in Figure 13. The accuracy of the training increases from ~79.5% to 88.2% and the training loss reduces from 0.53 to 0.34.

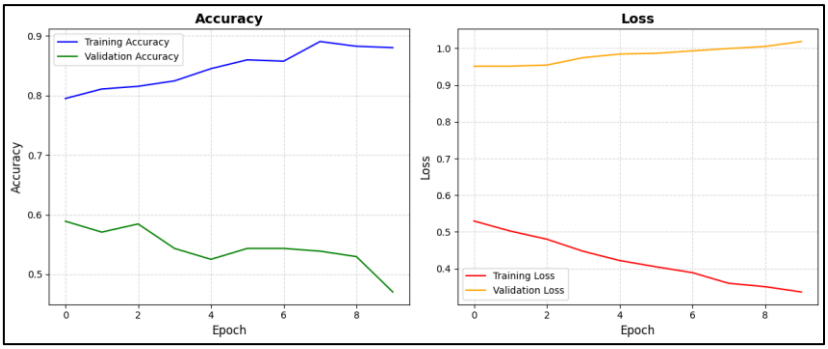

Figure 13. Accuracy Loss Curve Graph of CNN + PSO+ GA+ SA

The validation accuracy range is between 59.0% and 47.0% while the validation loss range is 0.95 to 1.02, which suggests that the model has a good learning ability but it is observed that there is a little overfitting in the later training epochs.

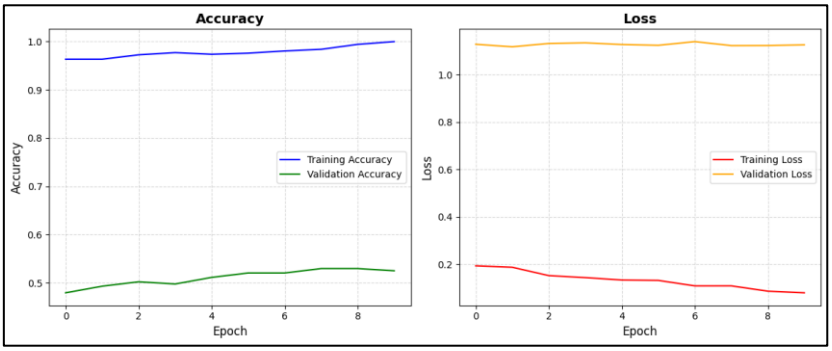

Figure 14. Accuracy Loss Curve Graph of MobileNetV2 + PSO+ GA+ SA

The training and validation accuracy and loss curves for the hybrid PSO–GA–SA optimized MobileNetV2 model trained for 10 epochs are shown in Figure 14.The accuracy and loss curves of hybrid PSO–GA–SA optimized MobileNetV2 model are presented in Figure 14 over 10 epochs of training and validation. The accuracy of the training process rises from about 96.3% to 99.8% and the training loss becomes 0.19 and 0.08 respectively. The accuracy of the validation set is marginally improved from 48.0% to 52.5% while loss on the validation set is almost constant around 1.12–1.14 with excellent training convergence but with a limited level of generalization and a high level of overfitting.

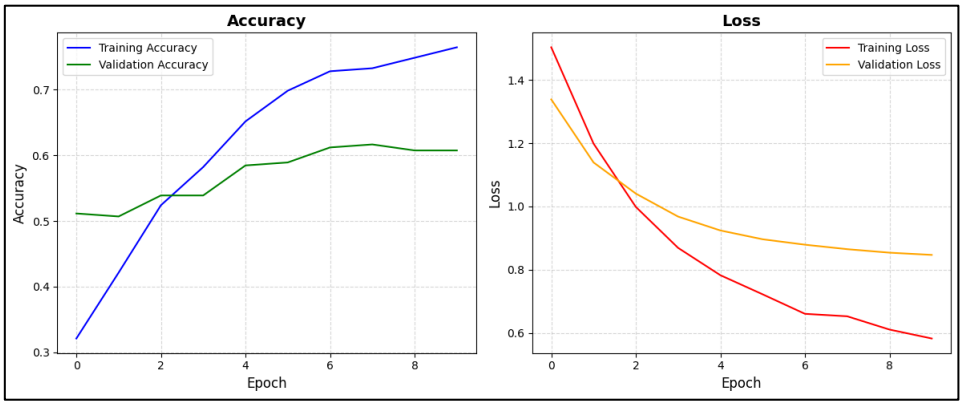

Figure 15. Accuracy Loss Curve Graph of DenseNet121 + PSO+ GA+ SA

The accuracy and loss curves for the hybrid PSO–GA–SA optimized DenseNet121 model during training and validation are shown, respectively in figure15(a) and figure15(b). The accuracy of the training increases from around 32% to 76% and the loss in training also decreases from 1.50 to 0.58. The accuracy of validation increases from 51% to 61%, while the loss of validation decreases from 1.34 to 0.85, showing that it walks in a stable path in convergence, decreases the optimization error and improves the generalization.

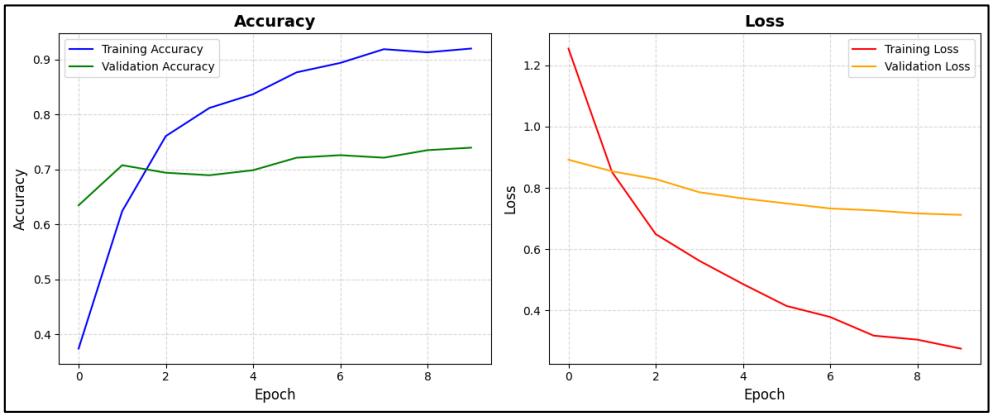

Figure 16. Accuracy Loss Curve Graph of InceptionV3 + PSO+ GA+ SA

The accuracy and loss curves during training and validation of the hybrid PSO–GA–SA optimized inceptionV3 model over 10 epochs are shown in figure 16. The accuracy of the training rises from ~37% to 92% and the training loss falls from 1.25 to 0.28. The accuracy of validation increases from 63% to 74%, while the loss of validation decreases from 0.89 to 0.71, showing that it is stably converging, optimising effectively, and improving the performance of multiclass lung cancer classification.

G. Comparative Analysis

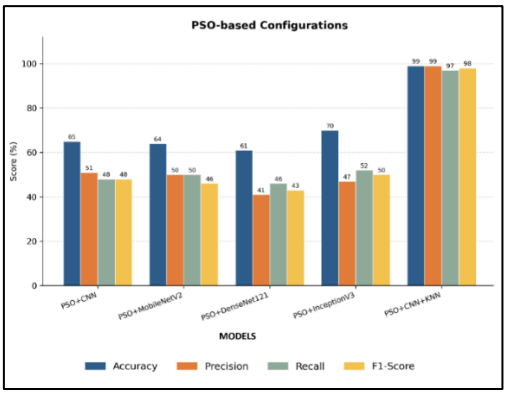

Figure 17. Performance Comparison of PSO-Optimized CNN-Based Classification Models for Lung Cancer Detection

The performance of the classification models based on CNN that were optimized by PSO for lung cancer detection is compared in figure 17. Hybrid CNN+KNN model gives the best accuracy of 99%, precision of 99%, recall of 97% and also gives an f1 score of 98%. On the other hand, InceptionV3 has an accuracy rate of 70%, and CNN, MobileNetV2 and DenseNet121 have an accuracy rate of 65%, 64% and 61%, respectively, proving the superiority of the PSO-optimized hybrid framework.

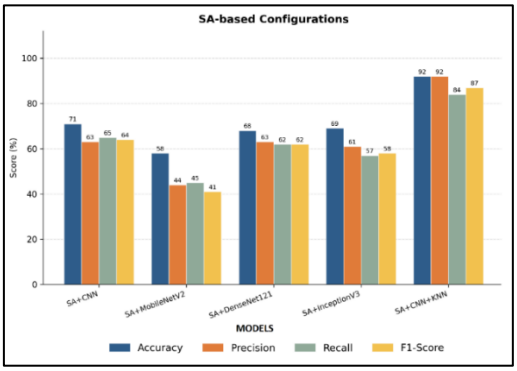

Figure 18. Comparative Performance of SA-Based Deep Learning Classification Models

It is seen that in figure 18, among all deep learning classification models based on SA, the maximum accuracy is achieved by CNN model. The hybrid CNN+KNN model obtains the best results with an accuracy of 92%, a precision of 92%, a recall of 84% and an F1 score of 87%. In terms of standalone models, CNN achieves the highest accuracy score (71%), followed by InceptionV3 (69%), DenseNet121 (68%) and MobileNetV2 (58%), showing the power of the hybrid solution designed for the SA.

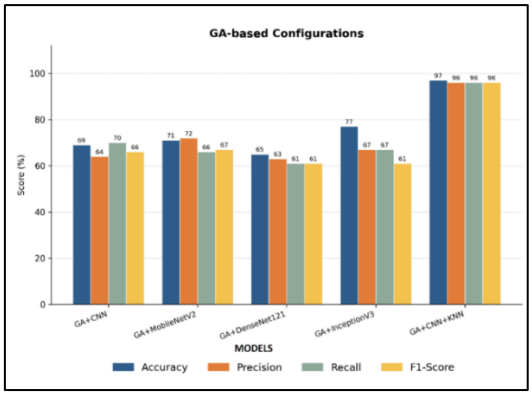

Figure 19. Comparative Performance of GA-Based Deep Learning Classification Models

The performance of the deep learning classification models based on GA is compared with each other in lung cancer detection as shown in figure 19. The hybrid CNN+KNN model outperforms all other models in terms of accuracy, precision, recall and F1 score with 97%, 96%, 96% and 96%, respectively. The GA-based optimization algorithm has worked very well in multiclass classification and achieved highest accuracy of 77% among individual models, followed by MobileNetV2 (71%), CNN (69%) and DenseNet121 (65%).

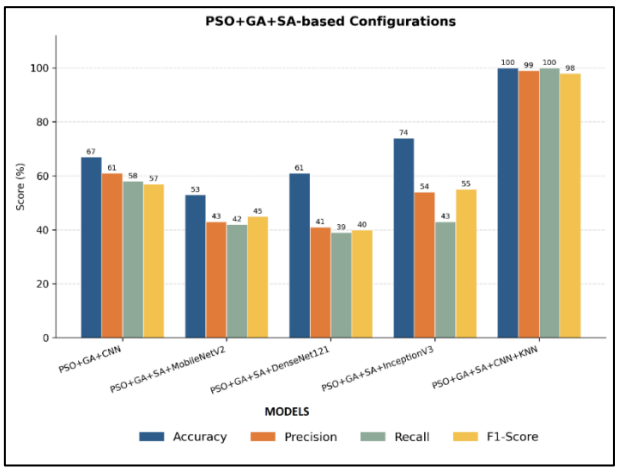

Figure 20. Performance Comparison of Hybrid PSO–GA–SA Optimized CNN-Based Classification

A comparison of the performance of these lung cancer detection models optimized using hybrid PSO–GA–SA algorithm is shown in figure 20. CNN+KNN model is found to be the best with 100% accuracy, 99% precision, 100% recall and 98% F1 Score. The standalone models perform below par, with InceptionV3 having the highest accuracy (74%), then CNN (67%), DenseNet121 (61%) and MobileNetV2 (53%). This is an indication of the better performance of the hybrid optimization approach.

Table 2. Comparative Performance Analysis of CNN-Based Classification Models under PSO, GA, SA, and Hybrid PSO–GA–SA Hyperparameter Optimization

Optimization Method	Classification Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
PSO	CNN	65	51	48	48
PSO	MobileNetV2	64	50	50	46
PSO	DenseNet121	61	41	46	43
PSO	InceptionV3	70	47	52	50
PSO	CNN+KNN	99	99	97	98
GA	CNN	69	64	70	66
GA	MobileNetV2	71	72	66	67
GA	DenseNet121	65	63	63	61
GA	InceptionV3	77	67	67	61
GA	CNN+KNN	97	96	96	96
SA	CNN	71	63	65	64
SA	MobileNetV2	58	44	45	41
SA	DenseNet121	68	63	62	62

SA	InceptionV3	69	61	57	58
SA	CNN+KNN	92	92	84	87
PSO+GA	CNN	67	61	58	57
PSO+GA+SA	MobileNetV2	53	43	42	45
PSO+GA+SA	DenseNet121	61	41	39	40
PSO+GA+SA	InceptionV3	74	54	43	55
PSO+GA+SA	CNN+KNN	**100**	**99**	**100**	**98**

The comparative performances of the CNN-based classification models optimized by PSO, GA, SA and PSO–GA–SA algorithms is shown in Table 2. The PSO–GA–SA optimized CNN–KNN hybrid model had the best overall performance of all the tested models with 100% accuracy, 99% precision, 100% recall and 98% F1-score, thus demonstrating a better predictive power and classification balance. Figure 21 shows that the accuracy and robustness of the hybrid PSO–GA–SA optimization strategy are superior in the classification performance.

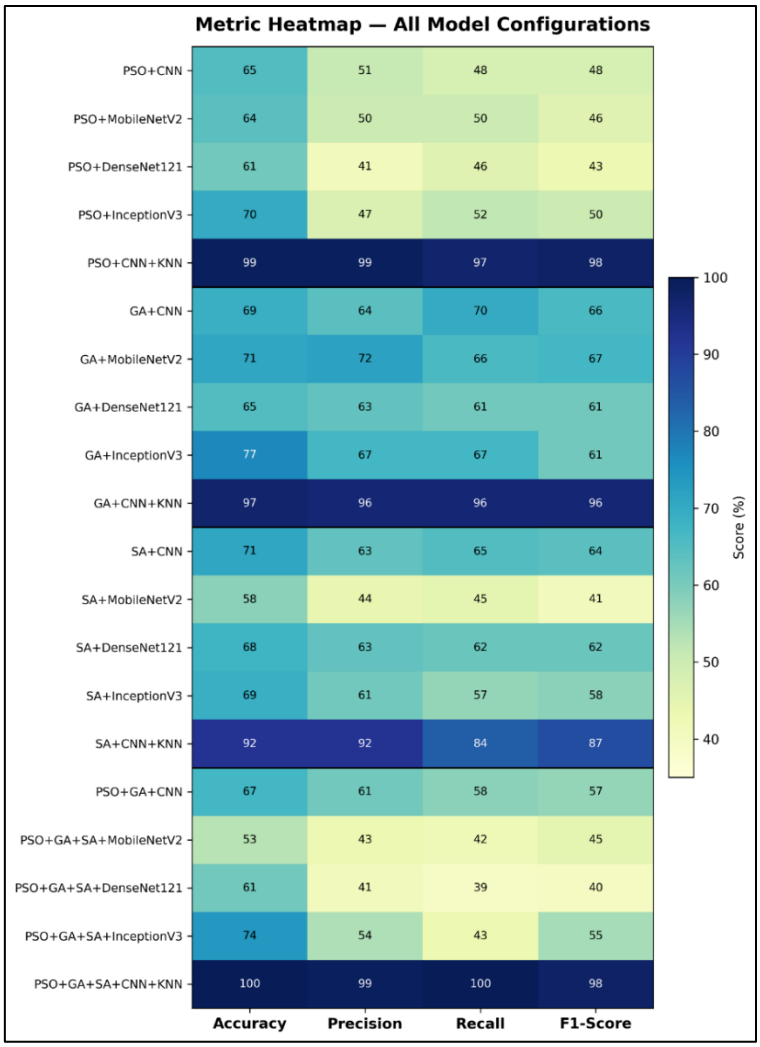

Figure 21. Visualization of Classification Performance across Optimization Strategies

Moderate improvement was achieved using individual optimization methods, GA being mostly superior to PSO and SA when applied to stand-alone deep learning models. As shown in Fig. 22, hybrid PSO–GA–SA always performs better at the multiclass classification task when compared to the other deep learning models evaluated.

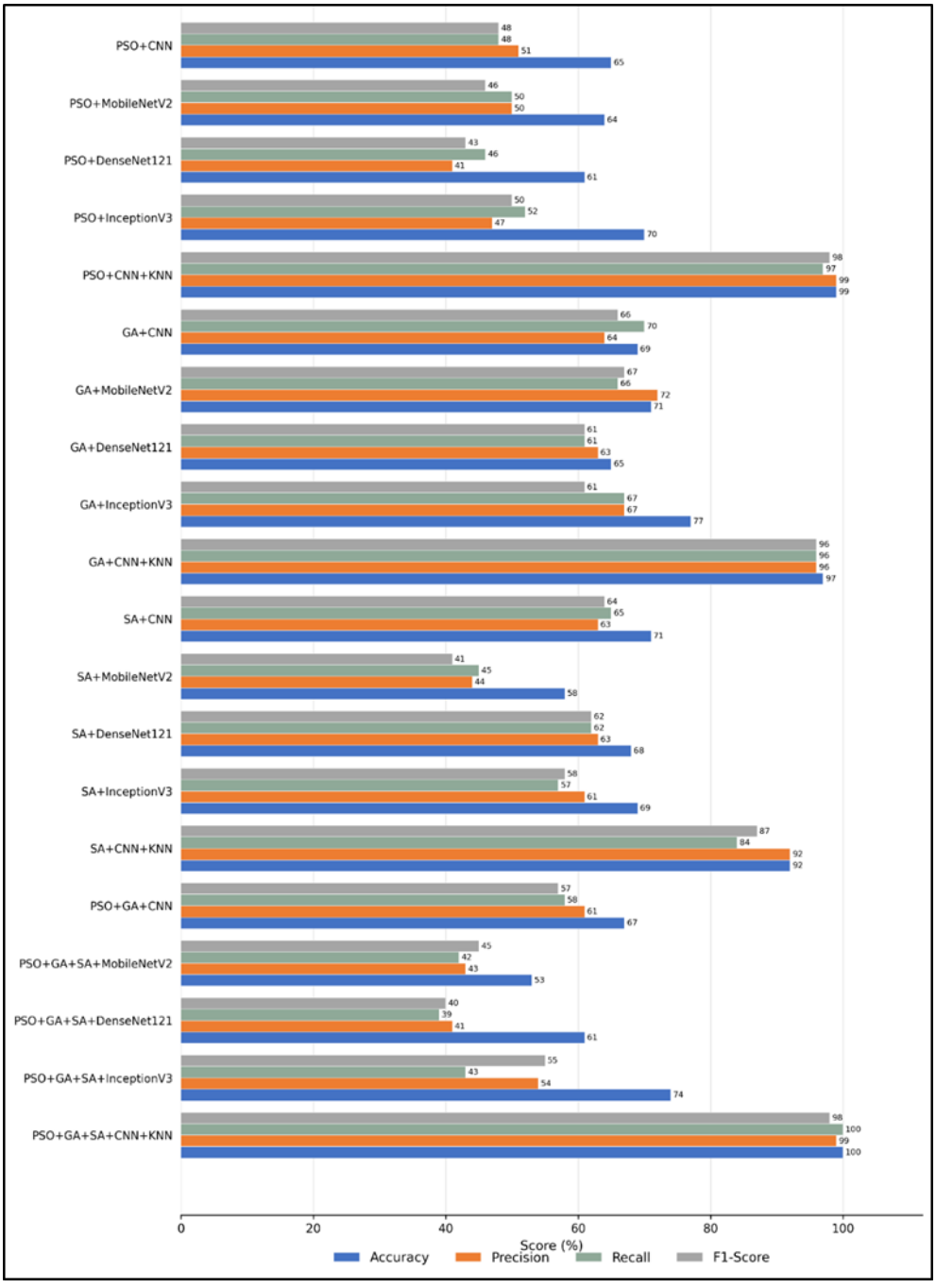

Figure 22. Overall Comparative Analysis of Deep Learning Classification Models across Optimization Strategies

The CNN–KNN hybrid always outperforms CNN, MobileNetV2, DenseNet121, and InceptionV3 for all of the optimization strategies, which proved the robustness of the CNN–KNN hybrid optimization and feature-based classification for automated multiclass lung cancer detection.

5. CONCLUSION

This study proposed a hybrid PSO–GA–SA based hyperparameter optimization system with CNN–KNN based multi-class lung cancer classification system, which is capable of automatically classifying lung cancer into four classes from computed tomography (CT) images. This new method integrated the global search ability of PSO, the evolutionary search ability of GA and the local search ability of SA to find optimal hyperparameters for network. Experimental tests showed that the hybrid optimisation approach was superior to the individual optimisation strategies and the stand-alone deep learning models at all times. The optimized CNN–KNN framework had the highest accuracy of 100%, precision of 99%, recall of 100%, and F1 score of 98% predicting its strong predictive ability and robustness. The hyperparameters were also optimized to increase the convergence and decrease overfitting by using optimal filter selection, layer density, dropout and learning rate optimization. Comparative analysis, confusion matrix and training–validation performance were used as further validations of the effectiveness and stability of the proposed model. In summary, the developed framework is a reliable and efficient decision-support system that could be utilized for early lung cancer diagnosis applications in the clinical environment and could serve as a foundation for future clinical screening applications based on AI.

REFERENCES

[1] Abdullahi, K., Ramakrishnan, K., & Ali, A. B. (2025). Deep Learning Techniques for Lung Cancer Diagnosis with Computed Tomography Imaging: A Systematic Review for Detection, Segmentation, and Classification. Information, 16(6), 451. https://doi.org/10.3390/info16060451

[2] Ansari, G. A., Shafi, S., & Alhazzaa, L. (2026). Optimized Machine Learning Pipeline for Lung Cancer Classification: Feature Reduction and Hyperparameter Tuning. Diagnostics, 16(8), 1198. https://doi.org/10.3390/diagnostics16081198

[3] Hewa, A., Razmara, J., & Karimpour, J. (2026). A Hybrid HOG-LBP-CNN Model with Self-Attention for Multiclass Lung Disease Diagnosis from CT Scan Images. Computers, 15(2), 93. https://doi.org/10.3390/computers15020093

[4] Demiröz, A., & Aydın Atasoy, N. (2025). Explainable Model of Hybrid Ensemble Learning for Prostate Cancer RNA-Seq Classification via Targeted Feature Selection. Electronics, 14(20), 4050. https://doi.org/10.3390/electronics14204050

[5] Yang, X., Duan, A., Jiang, Z., Li, X., Wang, C., Wang, J., & Zhou, J. (2026). Segmentation and Classification of Lung Cancer Images Using Deep Learning. Applied Sciences, 16(2), 628. https://doi.org/10.3390/app16020628

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

[7] M S, K., Rajaguru, H., & Nair, A. R. (2024). Enhancement of Classifier Performance with Adam and RanAdam Hyper-Parameter Tuning for Lung Cancer Detection from Microarray Data—In Pursuit of Precision. Bioengineering, 11(4), 314. https://doi.org/10.3390/bioengineering11040314

[8] Mondol, P. K., Islam Mozumder, M. A., Cheol Kim, H., Hassan Ali Al-Onaizan, M., Hassan, D. S. M., Al-Bahri, M., & Muthanna, M. S. A. (2025). A ResNet-50–UNet Hybrid with Whale Optimization Algorithm for Accurate Liver Tumor Segmentation. Diagnostics, 15(23), 2975. https://doi.org/10.3390/diagnostics15232975

[9] Guido, R., Ferrisi, S., Lofaro, D., & Conforti, D. (2024). An Overview on the Advancements of Support Vector Machine Models in Healthcare Applications: A Review. Information, 15(4), 235. https://doi.org/10.3390/info15040235

[10] Chakra Bortty, J., Chakraborty, G. S., Noman, I. R., Batra, S., Das, J., Bishnu, K. K., Tarafder, M. T. R., & Islam, A. (2025). A Novel Diagnostic Framework with an Optimized Ensemble of Vision Transformers and Convolutional Neural Networks for Enhanced Alzheimer's Disease Detection in Medical Imaging. Diagnostics, 15(6), 789. https://doi.org/10.3390/diagnostics15060789

[11] Chowdhuryan, H. (2026). Smart Healthcare Monitoring and Patient Tracking Using Neural Wireless Systems. International Journal on Advanced Computer Theory and Engineering, 15(2), 40–45. https://doi.org/10.65521/ijacte.v15i2.3303

[12] Bhakar, S., Sinwar, D., Pradhan, N., Dhaka, V. S., Cherrez-Ojeda, I., Parveen, A., & Hassan, M. U. (2023). Computational Intelligence-Based Disease Severity Identification: A Review of Multidisciplinary Domains. Diagnostics, 13(7), 1212. https://doi.org/10.3390/diagnostics13071212

[13] Almars, A. M., Alwateer, M., Qaraad, M., Amjad, S., Fathi, H., Kelany, A. K., Hussein, N. K., & Elhosseini, M. (2021). Brain Cancer Prediction Based on Novel Interpretable Ensemble Gene Selection Algorithm and Classifier. Diagnostics, 11(10), 1936. https://doi.org/10.3390/diagnostics11101936

[14] Nguyen, L. T. H., Shirai, T., & Tsuji, T. (2025). Characterization of Rock Pore Geometry and Mineralization Process via a Random Walk-Based Clogging Scheme. Algorithms, 18(2), 68. https://doi.org/10.3390/a18020068

[15] Pande, P. K., Khobragade, P., Ajani, S. N., and Uplanchiwar, V. P., "Early Detection and Prediction of Heart Disease with Machine Learning Techniques," in Proc. 2024 Int. Conf. Innov. Challenges Emerging Technol. (ICICET 2024), 2024, doi: 10.1109/ICICET59348.2024.10616294.

[16] Ji, R., Sorosh, S., Lo, E., Norton, T. J., Driscoll, J. W., Kuester, F., Barbosa, A. R., Simpson, B. G., & Hutchinson, T. C. (2025). Application Framework and Optimal Features for UAV-Based Earthquake-Induced Structural Displacement Monitoring. Algorithms, 18(2), 66. https://doi.org/10.3390/a18020066

[17] Ben-Mizrahi, Y. (2026). Dual-Channel Neural Intelligence for Healthcare Localization in Assisted Living Environments. International Journal on Advanced Computer Engineering and Communication Technology, 15(2), 41–47. Retrieved from https://journals.mriindia.com/index.php/ijacect/article/view/3377

[18] Abdelfattah, W. M. (2025). A Comparative Study of Differential Quadrature Methods for METE Nanobeam Vibrations. Algorithms, 18(2), 64. https://doi.org/10.3390/a18020064

[19] Gameiro, T., Pereira, T., Moghadaspoura, H., Di Giorgio, F., Viegas, C., Ferreira, N., Ferreira, J., Soares, S., & Valente, A. (2025). Evaluation of PID-Based Algorithms for UGVs. Algorithms, 18(2), 63. https://doi.org/10.3390/a18020063

[20] Dronyuk, I. (2025). Algorithms for Calculating Generalized Trigonometric Functions. Algorithms, 18(2), 60. https://doi.org/10.3390/a18020060

[21] Mariettou, S., Koutsojannis, C., & Triantafillou, V. (2025). Artificial Intelligence and Algorithmic Approaches of Health Security Systems: A Review. Algorithms, 18(2), 59. https://doi.org/10.3390/a18020059

[22] Wang, C., Shi, L., & Luo, J. (2025). Adaptive Noise Exploration for Neural Contextual Multi-Armed Bandits. Algorithms, 18(2), 56. https://doi.org/10.3390/a18020056

[23] Wakhare, P., Sheikh, S. M. S., Mahale, P., Patil, P., Bhilegaonkar, S., & Gulhane, M. (2025). AI-driven drug resistance profiling in tuberculosis patients: A transfer learning approach. Indian Journal of Tuberculosis.

[24] Shimbre, Nivedita, and Ram Kumar Solanki. "ChestXFusionNet: A multimodal deep learning framework for predicting chest diseases from X-ray images and clinical data." EPJ Web of Conferences. Vol. 328. EDP Sciences, 2025.

[25] Mane, D., Ashtagi, R., Suryawanshi, R., Kaulage, A. N., Hedaoo, A. N., Kulkarni, P. V., & Gandhi, Y. (2024). Diabetic Retinopathy Recognition and Classification Using Transfer Learning Deep Neural Networks. *Traitement du Signal*, *41*(5), 2683.

[26] Manop Phankokkruad. 2021. Ensemble Transfer Learning for Lung Cancer Detection. In 2021 4th International Conference on Data Science and Information Technology (DSIT 2021). Association for Computing Machinery, New York, NY, USA, 438–442. https://doi.org/10.1145/3478905.3478995