Journal of Theoretical and Applied Information Technology
30th November 2024. Vol.102. No. 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

OPTIMIZED EFFICIENTNET WITH GENETIC EXPRESS PROCESSING FOR ACCURATE LUNG DISEASE CLASSIFICATION

YELLEPEDDI SAMBA SIVA KRISHNA ASSISH[1] AND KUPPUSAMY P[1*]

[1]School of Computer Science and Engineering, VIT-AP University, Andhra Pradesh 522237, India

yellepeddi.22phd7108@vitap.ac.in[1], drpkscse@gmail.com[1*]

ABSTRACT

The ongoing COVID-19 pandemic underscores the urgency for rapid and precise diagnostic tools. This study presents an innovative approach for classifying lung diseases COVID-19, asthma, and pneumothorax using Computed Tomography (CT) lung images. The EfficientNet B4 is proposed to classify lung diseases accurately using compound scaling features including depth, width, and resolution. By integrating the EfficientNet model with a Genetic Express Processing Algorithm (GEP) for hyperparameter tuning, the proposed method focuses on optimizing dropout, learning rate, and batch size. Fine-tuning the EfficientNet B4 model through compound scaling and hyperparameter optimization led to a classification accuracy of 96.5%. Visualizing lung-infected regions using Class Activation Maps (CAMs) provides insights into classification decisions. This research work incorporates Generative Adversarial Networks (GANs) to generate synthetic images that enhances data diversity and model generalization. This method combines Deep Learning (DL) models with Genetic Algorithms (GA) and GANs, demonstrating substantial improvements in disease detection accuracy. The proposed approach offers medical professionals efficient diagnostic tools for early and reliable disease diagnosis. Code can be available at https://github.com/YellepeddiSambaSivaKrishnaAssish/Optimized-EfficientNet-using-GEP-for-Lung-diseases.git.

Keywords: *Computed Tomography, COVID-19, Genetic Express Processing, Artificial Intelligence, Optimization, Deep Learning*

1. INTRODUCTION

In December 2019, Wuhan, Hubei, China, became the epicenter of a COVID-19 pneumonia outbreak that rapidly spread worldwide. In the absence of a therapeutic vaccine or targeted antiviral medications, early diagnosis, and isolation became essential safety measures against the virus. The COVID-19-infected people undergone for scanning to identify the disease. However, the effectiveness of using chest CT scans to screen patients suspected of having COVID-19 was unclear [1]. To enhance understanding of clinical and radiologic symptoms of the infection, several academic studies, case reports, and sample analyses were published in the lateral weeks [2]. Large datasets are crucial for training DL models to avoid overfitting and bias. Researchers faced significant challenges in acquiring publicly available data and improperly collected data that did not meet the standards required for use in classification tasks. However, the limited availability of public data in the healthcare sector created a significant barrier for researchers. To address this, the DL community proposed Data Augmentation techniques to expand the size of training datasets [3]. Currently, highly skilled medical professionals manually evaluate CT and Chest X-ray (CXR) images and recommend treatments to radiological analyses, including chest CT scans and CXR, have demonstrated high accuracy in diagnosing COVID-19, with a strong correlation observed between RT-PCR results and radiological data. This has led to the development of faster and more cost-effective radiological COVID-19 screening methods [4]. To reduce infection levels, fortunately, the process is labor-intensive and time-

consuming, enhancing the need for a DL-based automated approach. By leveraging automated systems, it is possible to improve diagnostic efficiency, optimize the use of medical resources, reduce the workload on healthcare professionals, conserve their energy, and increase the accuracy and speed of diagnoses [5].

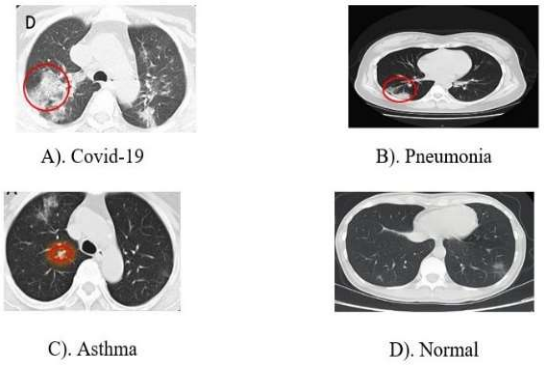

Figure 1. Images in Dataset

The above Figure 1. shows the images of four classes in the dataset. These images are utilized in our model to assess its efficiency. Class labels in both the train and test set are label-encoded into 0, 1, 2, and 3 for asthma, Covid-19, Pneumonia, and Normal. Otherwise, it can lead to weakness and post-traumatic stress. It's important to remember that distinguishing symptoms caused by COVID-19 from those of other conditions, such as preexisting medical issues, can be challenging. Therefore, it's crucial to diagnose COVID-19 effectively so that people can be alerted to its presence [6]. Recent technological advances have driven substantial research into automating the computer-aided diagnosis of COVID-19 and creating interactive tools to support the care and recovery of affected individuals. The adoption of automated methods offers several benefits, including reducing subjectivity, enhancing diagnostic availability and consistency, and enabling early detection [7]. This study outlines four key contributions aligned with the research objectives. The contributions made in this study are as follows:

- To improve image quality during preprocessing, GAN generates high-quality synthetic images at a resolution of 380 × 380, closely resembles real data, enhancing the classification model's performance.
- To optimize the EfficientNet classifier, fine-tuning the hyperparameters such as learning rate, dropout, and crossover enhances the model's accuracy and efficiency for lung disease classification.
- To Analyze and validate the empirical and experimental results of the proposed EfficientNet with GEP model across various medical imaging modalities, including CT scans, the model effectively showcases its ability to detect and classify lung diseases such as asthma, pneumothorax, COVID-19, and normal cases.
- Investigate and compare the existing model with the EfficientNet with the GEP model to highlight the importance of GEP in enhancing overall classification accuracy. This integrated pipeline, in conjunction with the EfficientNet with GEP model, enables accurate assessment of lung disease severity, offering critical insights to support medical decision-making.

Motivation

In the fight against global health crises such as COVID-19, rapid and accurate diagnosis is crucial, often determining life-or-death outcomes. In the healthcare community under immense pressure, there's an urgent need for smarter, faster tools to detect lung diseases from medical images. This research taps into the power of cutting-edge AI, using DL models like EfficientNet B4 and the precision of GA, to revolutionize lung disease diagnosis. Our objective is to arm healthcare professionals with powerful tools that ensure swift, accurate interventions, saving lives and advancing the fight against respiratory diseases.

Recently authors proposed a method that leverages contrastive learning, a technique that maximizes data utility by enhancing feature extraction and generalization, thereby improving classification accuracy compared to other methods. The Contrast Learning Framework (CLF) in CT-scan successfully identified COVID-19 cases from normal samples and various kinds of traditional pneumonia, which is critical for quick clinical screening. Furthermore, CLF is a full, deployable framework that includes data pretreatment, improvement, and classification

operations, enabling rapid generalization to various data domains.

2. RELATED WORK

Previous researchers have looked into using DL techniques to solve the difficulties associated with COVID-19 detection. To explore the potential of CT images in COVID-19 diagnosis, this section surveys existing literature on the work.

2.1. Deep Learning Strategies for CXR & CT-Scans in Computer-Aided Diagnosis (CAD)

Utilizing the temporal component Loey et al. executed a CNN model to extract important COVID-19 traits, and then a Bayesian model to classify patients according to temporal features [8]. Reddy et al. comprise three primary segments for the Multimodal Fusion Deep Transfer Learning (MMF-DTL) model, pre-processing, feature extraction, and classification, it uses three DL models: ResNet 50, Inception v3, and VGG16. The model uses decision-based multimodal fusion to integrate these three methods to improve detection efficiency [9]. Further contributions by researchers examine the effectiveness of GA and Binary Particle Swarm Optimization (BPSO) in enhancing Machine Learning (ML) classifiers for COVID-19 Disease data. By optimizing feature subsets, GA-inspired MLP achieved 85.1% accuracy with 52.32% dimensionality reduction, and the BPSO-inspired algorithm reached 90.7% accuracy with 41.43% reduction [10]. Despite the limitations of traditional diagnostic methods, a higher detection rate can be achieved with this combination of DL models with CAD for CXR & CT scans. In the end, test images are divided into six groups using a Softmax classifier. The model's execution is thoroughly examined using the CXR dataset, which demonstrates that it can correctly diagnose COVID-19 using imaging data. The metrics measured with an F-score of 93.26% and an average accuracy of 98.80%, the fusion model performs well. For COVID-19, Ayyar et al. [11] developed a hierarchical classification model using CXR images. The global attention mechanism's integrated DL model outperforms the baseline of the COVID-Net model, and it comprises multiple binary classification models. Researchers discovered that the COVIDx chest imaging dataset, which was recently the COVID-Net Open-Source Initiative has established one of the largest databases of freely accessible COVID-19-positive cases available. Furthermore, COVID-Net [12] was recommended as the COVIDx standard model, with a COVID-19 detection sensitivity of 91%.

To highlight COVID-19 diagnosis, the authors provide a unique multi-scale attention network that combines improved with original CXR lung images. It shows good generalization capabilities when tested on a variety of datasets [13]. The research provides a unique framework that combines ML and DL techniques to classify using CXR images of the chest to diagnose lung illnesses. It evaluates the effectiveness of several soft computing methods and feature normalization approaches in detecting COVID-19 and pneumonia. The work emphasizes how DL models and rigorous feature normalization are important for increasing lung disease classification accuracy [16]. The suggested article presents DeepCov19Net, an automated COVID-19 disease detection technique using DL. It makes use of an SVM classifier with different kernel functions for evaluation, a novel feature selection approach called SDAR, and a convolutional-autoencoder model for deep feature extraction. The proposed approach correctly categorized CXR images for COVID-19, normal, and pneumonia patients [17].

Table 1. Existing research works

Author Name	Method Name	Dataset	Performance (%)	Limitations
Loey et al. [8]	Bayesian optimized CNN	CXR	Accuracy 96	Dataset is imbalanced
K.N. Bhramaji et al. [9]	Multi-modal fusion deep transfer learning (MMF-DTL)	CXR	Average Sensitivity 93, Precision 90, Accuracy 98, kappa 91	The usage of large datasets, so can improve the training process and also improve the detection rate.

Meghana, Jenny et al. [10]	Multi-level Classification Covid-NET (Resnet-50, inception V3)	CXR	Accuracy 91, Sensitivity 95	The results can still be optimized by the optimization techniques.
Linda, Zhong Qui et al [11]	Covid-NET	CXR	Accuracy 90, Precision 92	The model can be optimized and integrated with other diagnostic tools.
Goyal and Rajiv [12]		CXR		
Fares and Cosimo et al [13]	Resnet-50, Densenet-161 and inception V3	COVID-CT	Accuracy 88, Sensitivity 95	To improve classification accuracy the dataset should be balanced.
Vruddhi Shah, Rinkal et al [14]	CTnet-10 VGG-19	COVID-CT	Accuracy 94	Data scarcity restricts model effectiveness for all patients
Parnian, Shahin et al [15]	Fully automatic Capsule Network	COVID-CT	Accuracy 90.8, Sensitivity 94.5, and specificity 86.0	Image enhancement is required and Requires validation on larger, multi-center datasets for robustness.
Godbin and jasmine [20]	Support Vector Machine (SVM)	Sars COV2-CT	Accuracy 94.9	The model can be tested on deep learning classifiers and even on various datasets can be performed
Ravi Lakshmi et al [21]	Customized CNN Image channel model	Sars Cov-2 CT Covid-CT	Accuracy 89 Precision 90	Here it has a lot of scope for residual network and to increase the regularization layer
Esraa Hassan,y. shams et al [22]	Resnet-50, VGG-19, VGG-16, DCNN	COVID-X CT	Accuracy 96.23	It includes scalability issues from high computational demands, reliance on quality training data, and lack of real-world validation, affecting its clinical applicability.

Observations

Observations from Table 1. highlight key research efforts in using ML and DL for COVID-19 disease detection from CXR and CT scans. It provides various methods, detailing their performance metrics like accuracy and sensitivity, while also pointing out limitations such as data imbalance, scalability issues, and the need for further optimization. various COVID-19 detection models using CXR and CT datasets, with accuracy ranging from 87% to 98%. Common challenges include imbalanced datasets, data scarcity, and the need for optimization techniques [27-28]. Some models struggle with dataset-related issues, while others require image enhancement and broader validation. This concise review offers a quick comparison of different approaches, helping to identify the most promising models and areas for improvement in medical imaging diagnostics.

2.2. COVID-19 Classification

To quickly screen COVID-19, researchers considered noisy CXR images, and their robustness was evaluated by a combination of neural networks with transfer learned weights from AlexNet, DenseNet121, InceptionV3, resNet18, and GoogLeNet [18]. The model's accuracy of 95.83% was acceptable. The model could still be made more robust, although. AI-based chest CT imaging, with its extensive inspection range and variety of images, is also utilized in the diagnosis of viral pneumonia. Since COVID-19 has developed quickly, more DL techniques have become available to the wider public [19]. It is suggested that Covid-Vision-Transformers (CovidViT) be used to identify Covid-19 instances from CXR scans. Transformers block with the self-attention mechanism serve as the foundation for CovidViT. To prove its superiority, the experimental result demonstrates that CovidViT achieves 98.0% accuracy on the test set and surpasses other DL models [20].

By harsening AI capabilities Godbin et al. suggested employing ML models that use radiomics, such as GLCM characteristics from CT scans, to get 99.94% precision in evolving COVID-19 from comparable illnesses like pneumonia [21]. The scientists employed a Deep Convolutional Neural Network to categorize the COVID-19 and Normal CT scan inputs. Pre-trained transfer learning models used in the study include ResNet (50), VGG (19), VGG (16), and Inception V3. The binary cross-entropy metric, based on the anticipated likelihood for each class, is used to compare COVID-19 cases to normal ones. Overfitting problems are dealt with by using Adam optimizers and stochastic gradient descent [22]. Researchers offered a multi-level feature attention network (MLF-AttNet) consisting of a data analyst and a data preprocessor for COVID-19 identification. The six-layer data preprocessing denoises and augments the input health information, while the four-layer combined information analyzer extracts features by analyzing and classifying the source data's multi-scale, multi-level, and temporal dimensions. In the future, it can also look at unsupervised and semi-supervised DL methods to improve the competitiveness of our model [23].

The authors proposed a Squeeze-and-Excitation Network (SENet) that reweights feature representations and records channel-wise information. To direct networks' attention, this module can be freely inserted into any feature-extract layer. Different variants have been created to handle channel features even more effectively after SENet [24]. Previous studies have extensively explored hyperparameter optimization techniques. Two of the most popular searches are grid search and random search. While random search picks hyperparameters at random, grid search conducts an exhaustive search across a deliberately defined subset of the hyperparameter space. These techniques are time-consuming and computationally expensive, even if they can be somewhat effective, especially for large datasets and sophisticated models [25]. The exploration extends to COVID-19 diagnosis using CT scans and is promising but limited by the need for expert radiologists and variability in interpretations. A research study that used a dataset of 349 positive and 397 negative CT images to fine-tune 15 pre-trained CNNs for COVID-19 detection was introduced. In connection with accuracy (0.85), recall (0.854), and precision (0.857), the ensemble comprising EfficientNetB0, B3, B5, Inception_resnet_v2, and Xception is far better than any one of the individual models. This illustrates how deep transfer learning may be used to accurately diagnose COVID-19 [26].

3. METHODOLOGY

At this juncture, the article methodology is organized in a structure as follows: (i) Pre-processing techniques; (ii)EfficientNet B4 Architecture; (iii) Genetic Algorithm Evolution; and (iv) Empirical Analysis of GA.

3.1. Preprocessing techniques

The preliminary lung image processing is done to remove noise and abnormalities from the COVID-19 image to improve its contrast and quality. This pre-processing step can assist reduce inconsistencies and increase the accuracy of other stages in the process by improving contrast, reducing noise, and normalizing the image. Image processing attempts to improve and enhance the quality of COVID-19 CT scans in the pursuit of identifying clinical characteristics associated with lung illnesses. The Generative Adversarial Network (GAN) is feasible and can significantly enhance preprocessing for classifying diseases such as COVID-19, Pneumonia, Asthma, and Normal. Incorporating GANs into your lung disease classification task involves generating

synthetic images for each class (Covid-19, Pneumonia, Normal, and Asthma). These synthetic images, combined with traditional data augmentation techniques like rotation and flipping, can help balance the dataset, improve model generalization, and enhance classification accuracy by providing a more diverse and robust training set. The EfficientNet's advanced feature extraction capabilities can be leveraged within the GAN framework, either as part of the discriminator to improve feature differentiation or in the generator to produce high-quality images.

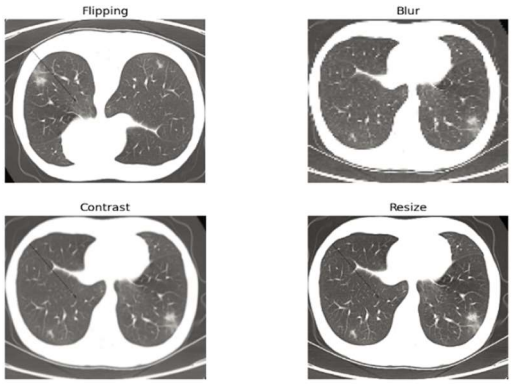

Figure 2. Techniques used during Preprocessing the Lung CT scans

Before Preprocessing, initially the lung images in the dataset are in different resolution ranges from 102×137 to 1853×1485 . While CT scans are typically stored in DICOM format, the images from the two databases used in this study are in PNG format, Figure 2. exhibits the operations used during preprocessing, images were scaled to 380×380 pixels in the direction of the image to match the input criteria for the proposed CNN architecture. The CT images were then standardized to a range of 0 to 1 to improve the stability of the CNN model.

The lung CT image is then subjected to the center crop transformation, which crops it in the center to produce a final size of 380×380 pixels for optimal performance. The cropping technique ensures that important portions of the image, including the lungs and their surroundings, are focused on and preserved.

3.2. EfficientNet B4 Architecture

This section discusses the transfer learning paradigm with the fine-tuned architecture of the EfficientNetB4 in detail. EfficientNet B4 is a cutting-edge CNN that aims to achieve excellent accuracy with fewer parameters than classic models. It applies a compound scaling strategy to balance the network's depth, breadth, and lung image resolution. Figure 3. illustrates EfficientNet B4 uses compound scaling to scale depth, width, and resolution simultaneously. Width scaling increases the number of channels, helping the model capture detailed features in lung images. Depth scaling allows it to learn complex patterns like lung abnormalities across multiple layers. Resolution scaling ensures high-resolution input images retain crucial details for better disease detection. Here scaling balance optimizes EfficientNet B4 for accurate and efficient lung disease classification.

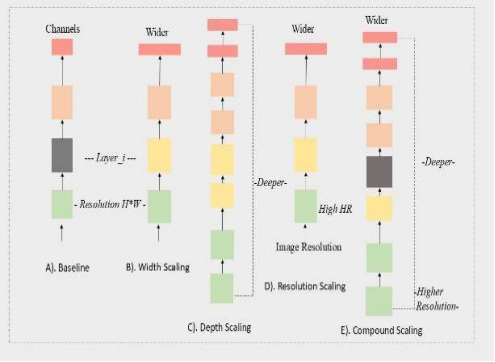

Figure 3. Illustration of Compound scaling of EfficientNet with Width, Depth & Resolution

Here, the pipeline illustration in Figure 4. will outline the architecture of EfficientNet B4 and its classification to classify lung diseases into four classes: asthma, COVID-19, pneumonia, and normal. The primary components include the stem layer, MBConv blocks, and Classification head.

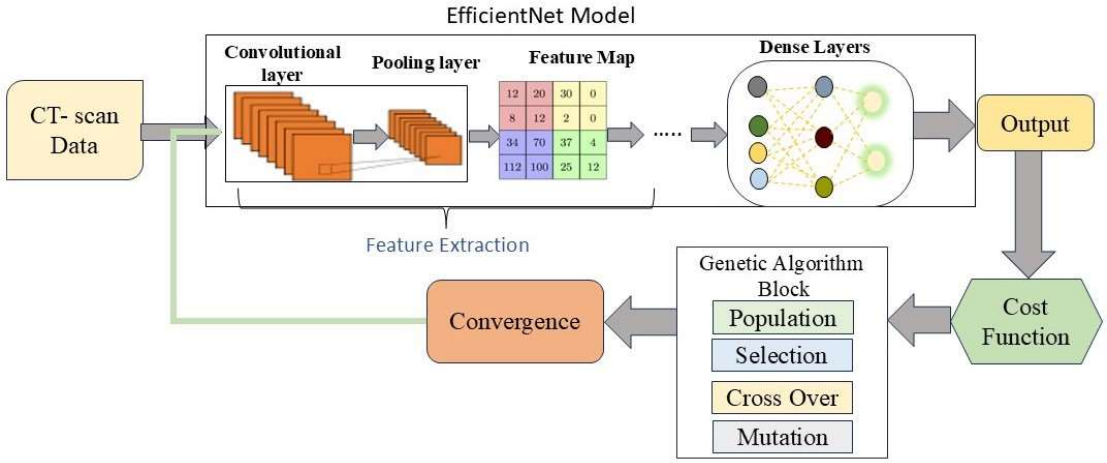

Figure 4. Pretrained EfficientNet with GA for classification

The stem layer is the initial layer that processes the input lung image, typically consisting of a convolutional layer followed by batch normalization and a Swish activation function. Mathematically, this can be represented as

$$Y = Swish(BatchNorm(Conv(X))) \quad (1)$$

Here, X is the input lung image, while Y is the stem layer's output. Following the input layer, the model includes several convolutional and pooling layers, organized into four main convolutional blocks. Each block is designed to progressively extract more complex features while reducing the spatial dimensions of the data.

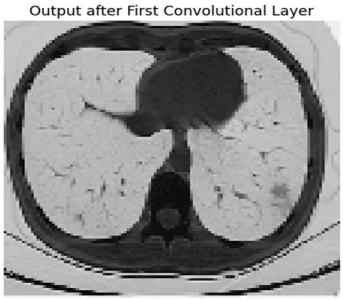

Figure 5. Output after the Convolutional layer

The first convolutional block can be visualized in Figure. 5 uses a Conv2D layer with 32 filters in total and a (3, 3) kernel size, followed by a Global Average Pooling (GAP) 2D layer with a pool size of (2, 2). The purpose of the GAP 2D layer is to reduce the dimensionality of the data, which helps in reducing the computational load and mitigating overfitting by discarding less important spatial information. The computational load on the GAP layer for lung images is minimal. GAP reduces the spatial dimensions of feature maps by averaging values across each channel to vector, resulting in fewer parameters and a lower risk of overfitting.

$$\text{Conv2D}(x, W, b) = W \otimes x + b \quad (2)$$

where ⊗ denotes the convolution operation, W is the filter (weight) matrix, x is the input, and b is the Bias term.

After the Convolutional layer, Batch Normalization is used to improve training efficiency and stability by minimizing internal covariate shift. The process involves four key steps: first, computing the mean μ_B of a mini-batch of inputs; second, calculating the variance σ_B^2 of the mini-batch; third, normalizing the inputs by removing the mean and dividing by the square root of the variance plus a tiny constant ϵ for numerical stability; and finally, scaling and shifting the normalized inputs using learnable parameters γ (scale) and β (shift). This normalization technique allows the network to maintain the representational power of the original activations while ensuring more stable and faster training.

$$y_i = \gamma\left(\frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}\right) + \beta \quad (3)$$

The above states where y_i is the output after batch normalization, x_i is the input, μ_B is the mean, σ_B^2 is the variance, ϵ is a small constant for numerical stability, and γ and β are learnable parameters.

The second convolutional block expands on the first by adding 64 filters while preserving the same kernel size and ReLU activation [31]. The Conv2D layer is followed by another GAP layer with a pool size of (2,2). This trend continues in the third convolutional block, where the Conv2D layer contains 128 filters and another GAP layer is applied. The fourth convolutional block also includes a Conv2D layer with 128 filters, followed by a GAP layer.

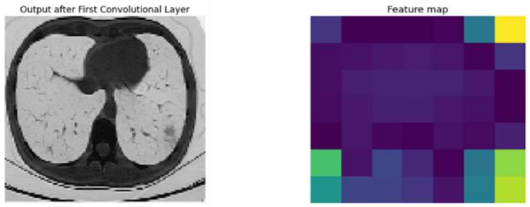

Figure 6. Output before the Dense layers

From the visualization of Figure 6. the initial layer captures basic features, while the final layer outputs a condensed feature map highlighting complex patterns. The processed lung image data is then used for classification in the dense layers, each of these blocks increases the network's depth, allowing it to record more detailed and abstract information as the input moves through the layers. Following a series of convolutional and pooling layers, the model progresses to fully connected (dense) layers. First, a flattened layer is utilized to turn the convolutional layers' 2D matrix data into a 1D vector. This transformation is crucial as it prepares the data for the dense layers, which operate on one-dimensional input.

$$y_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} F_c(i,j) \qquad (4)$$

The preceding equation describes the GAP, where H is the feature map's height, W is its width, F_c (i, j) is the feature map's value at position (i, j) for channel C, and y_c is the average value for the complete feature map F_c. The initial dense layer is made up of 512 units that use Softmax activation. This layer connects the high-level feature extraction achieved by the convolutional layers and the final classification task [32]. To stay away from overfitting, a 50% dropout layer is placed after the dense layer. Table 3. shows the layers of EfficientNet B4, which include a layer of convolution, batch normalization, global average pooling, dropout, and a dense layer.

Table 2. Layers Description

Layer	Output shape	Parameters
Input layer	()	0
Convolutional layer	(6,6,1536)	1404576
Batch Normalization	(6,6,1536)	6144
Global Average Pooling (GAP)	(1536)	0
DropOut	(1536)	0
Dense	(4)	6148

Dropout is a regularization strategy in which a portion of the units are randomly set to zero during training, forcing the network to learn more robust characteristics that are not dependent on any individual unit. The output layer is the model's final layer, and it is meant for classification. This layer is dense, containing as many units as there are classes in the classification problem. It employs softmax activation to generate a probability distribution over the classes, allowing the framework to predict which class the input image belongs to. The below equation is the softmax activation

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \text{ for i} = 1,2,\ldots\ldots,\text{k} \qquad (5)$$

Where $\sigma(z)_i$ represents the probability of the i-th class, e^{z_i} is the exponential of the input score of the i-th class, and $\sum_{j=1}^{k} e^{z_j}$ is the sum of the exponentials of the input scores for all classes.

The classification head is the final component, consisting of global average pooling, a fully connected (dense) layer, and a softmax activation to output the class probabilities. This can be mathematically described as

$$P = Softmax(Dense(GlobalAvgPool(F))) \qquad (6)$$

Here F is the feature map input and P provides the class probabilities for asthma, COVID-19, pneumonia, and normal. During training, the multi-label classification uses categorical cross-entropy loss. Here, the mathematical expression

$$L = -\sum_{i=1}^{N} y_i log(\hat{y}_i) \qquad (7)$$

Where y_i is the ground truth label and $\hat{y}_i$ is the predicted probability for class i. The Adam optimizer is used for training the model and updating model parameters as θ.

$$\theta_{t+1} = \theta_t - n.\nabla_\theta L(\theta_t) \quad (8)$$

Here n is the Learning rate. In this phase, the trained EfficientNet B4 model is used to predict the class probabilities for new input CT scans.

3.3. Genetic Algorithm Evolution

The GA is one of the ideal algorithms for addressing sophisticated optimization issues that are difficult to deal with employing conventional approaches [33,34]. It is a flowchart that illustrates the steps of a GA. Evolution and genetics provide as inspiration for GA search algorithms. They are used to identify the best answers to problems by creating a population of potential solutions via a process equivalent to biological reproduction.

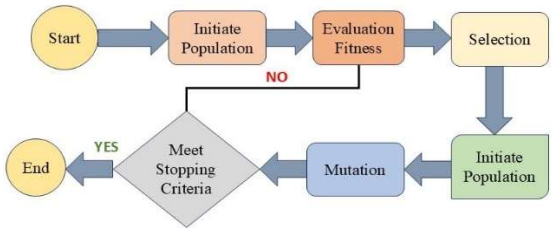

Figure 7. Workflow of GA

Algorithm: Efficient Net + GEP to optimize the hyperparameters of lung disease

```
procedure GEP_Classification (I, C)
// Input I = CT image for diagnosis
// Output C = predicted class for the image
// 1. Initialize population and evaluate initial fitness
for generation = 0; generation ≤ Max Generations; generation++ do
   // 2. Apply genetic operations for each individual in the population
   for i = 0; i ≤ Population Size; i++ do
      // a. Select parents for crossover
      Parent1, Parent2 ← Select Parent (Population, Fitness)
      Offspring ← Crossover (Parent1, Parent2)
      // b. Apply mutation to offspring
      Mutated Offspring ← Mutate (Offspring)
      // c. Evaluate fitness of mutated offspring
      Fitness Offspring ← Evaluate Fitness (Mutated Offspring)
      // d. Replace worst individual if offspring has better fitness
      if Fitness Offspring > Min Fitness then
         Population[i] ← Mutated Offspring
         Fitness[i] ← Fitness Offspring
      end if
   end for
end for
// 3. Select the best chromosome with highest fitness
Best Chromosome ← Select Best Chromosome (Population, Fitness)
C ← Predict Class (Best Chromosome)
return C
```

The Figure 7. flowchart starts with a baseline population of solutions. Each solution is evaluated using a fitness function, which determines how well it meets the desired criteria. In the selection step, individuals from the population are chosen based on their fitness. The individuals with greater fitness scores are more likely to be chosen. The algorithm applies crossover and mutation operators is done through an exchange or swap policy to the choose individuals. Crossover simulates reproduction by mixing the genetic information from two parent solutions and producing offspring.

The mutation causes random changes to individual solutions, which helps to preserve population diversity and prevents the algorithm from becoming stuck in local optima. The newly generated offspring are then evaluated using the fitness function. This cycle of evaluation, selection, crossover, and mutation will continue until the termination specifications are met. The termination criterion may include a predefined number of iterations or a desired level of fitness in the individuals.

$$pop = \begin{bmatrix} pop1 \\ pop2 \\ pop3 \\ \vdots \\ pop\ n \\ pop\ n-1 \end{bmatrix} \quad (9)$$

Furthermore, let's use the letter "pop" to stand for the population that consists of parents that have proven to have higher fitness values. When percentage P is set to 30%, only the fittest 30% of the individuals is considered suitable to take part in the crossover or recombination selection process. The F1 score measures the effectiveness of the model and is computed using the validation set. Hyperparameter tweaking is used to evaluate different parameter configurations, aiming to optimize the model's ability to distinguish between pneumonia and normal patients. To ascertain the ideal fitness value for subsequent utilization, see Equation.

$$ft = \sum_{K=1}^{M} R_s/M \quad (10)$$

The addition of $\frac{m}{K}$ to the summation term is represented by variable s, and certain parameters are the focus of improvement. In this case, M represents the entire number of regulations, and R_s indicates the rule that has been chosen. For every value, f_t is determined by the fitness value selected rules. Each value is evaluated in reference to the fitness function. Only solutions that align with the fitness function's requirements advance to either crossover or mutation-based generation. The purpose of mutation is to add unpredictability to the population by randomly altering a subset of hyperparameters [35-36]. Every solution in the population is prevented by the mutation from reaching the local optimum of the resolved problem. The result of crossing is an offspring population that is susceptible to random mutation

$$Mutation(C_i) = \begin{cases} 1 - C_i[k], & if\ r < p_m \\ C_i[k], & Otherwise \end{cases} \quad (11)$$

Here k refers to the features in the individual and r is the random number.

3.4. Empirical Analysis of GEP Evolution

The core of the architecture lies in the GA evolution process, which iterates through several generations to optimize the feature subset. The evolution process comprises multiple components, including classification, validation, and the generation of new populations through genetic operations. The empirical analysis demonstrates the effectiveness of GA in optimizing hyperparameters for a neural network. By iteratively evolving a population of candidate solutions, the GA converges towards better solutions, as evidenced by improved fitness values over generations. The GA workflow involves seven several steps

i). Initialization

The initial population consists of four individuals, each representing a set of hyperparameters. The initial population is randomly generated within the given ranges:

$$P(0) = \begin{Bmatrix} X_1=[0.001,0.5] \\ X_2=[0.01,0.3] \\ X_3=[0.005,0.4] \\ X_4=[0.02,0.2] \end{Bmatrix} \quad (12)$$

ii). Fitness Evaluation

Fitness is evaluated based on the validation accuracy of the neural network using the given hyperparameters. Higher accuracy indicates better fitness.

$$f(X_1) = 0.85,\ f(X_2) = 0.80, f(X_3) = 0.83,\ f(X_4) = 0.78$$

iii). Selection

Parents are selected using Roulette Wheel Selection, which is a probabilistic method where more fit individuals have an increased probability of selection. Assume X_2 and X_3 are selected as parents.

iv). Crossover

The chosen hyperparameters are learning rate and dropout so, the arithmetic crossover is performed with a parameter $\alpha = 0.6$ to attain offsprings.

$$Offspring_1 = 0.6.X_2 + 0.4.X_3 = [0.008,0.34]$$

$$Offspring_2 = 0.4.X_2 + 0.6.X_3 = [0.007,0.36]$$

V). Mutation

Here Gaussian mutation is applied with a standard deviation $\sigma = 0.001$ for Learning rate and $\sigma = 0.05$ for Dropout.

$$Mutated\ Offspring_1 = [0.0085,0.33]$$

$$Mutated\ Offspring_2 = [0.0068,0.38]$$

VI). Replacement

Here it includes the current population and new offspring then selecting the best individuals based on fitness.

$$P(1) = \begin{Bmatrix} X_1=[0.001,0.5]\ f(X_1)=0.85 \\ Mutated\ Offspring_1=[0.0085,0.33]\ f(Mutated\ Offspring_1)=0.84 \\ X_3=[0.005,0.4]\ f(X_3)=0.83 \\ Mutated\ Offspring_2=[0.0068,0.38]\ f(Mutated\ Offspring_2)=0.81 \end{Bmatrix} \quad (13)$$

4. PROPOSED ARCHITECTURE OF EFFICIENTNET WITH GENETIC EXPRESS PROCESSING

Genetic Express Processing (GEP) is an evolutionary algorithm [37]. For identifying lung diseases like COVID-19, asthma, pneumonia, and normal, GEP can effectively analyze complex medical data to distinguish between these

conditions. It incorporates the strength of GA and genetic programming to enhance classification accuracy. By leveraging GEP, robust models can be developed to improve diagnostic precision and patient outcomes.

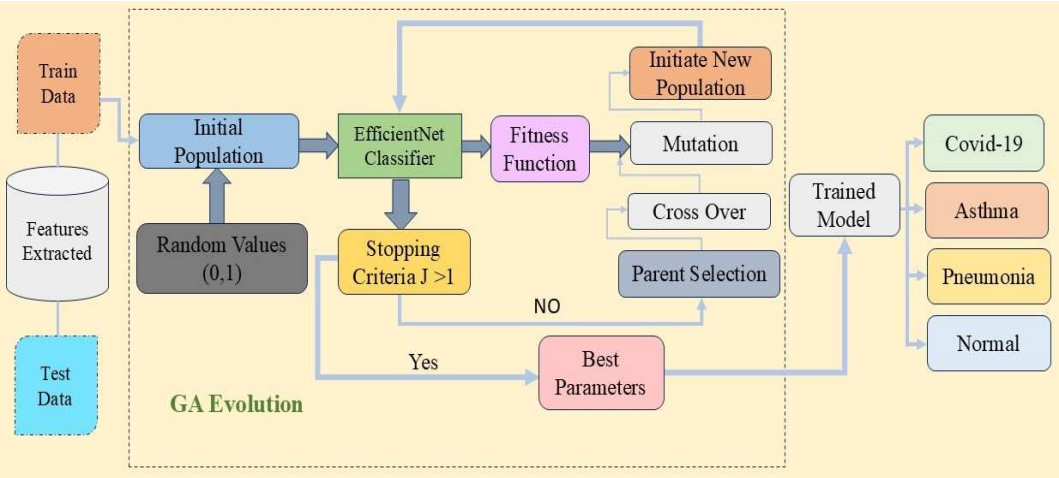

Figure 8. Schematic Representation of the proposed model using the GEP Algorithm

Figure 8. illustrates a bio-inspired optimization GEP workflow for classifying Covid-19, Asthma, Pneumonia, and Normal cases. It starts with extracting features from the training and test data. The train data's features are leveraged for model training, while test data's features evaluate its performance. In GEP, the GA evolution begins with creating an initial population of feature sets encoded with random binary values (0 and 1). These feature sets undergo classification, training, and validation.

Table 3. Hyperparameters for optimization

Hyper-Parameters	Values
Population	50
Generation	10
Drop Out	0.5
Batch size	32
Learning rate	0.001
Cross over	0.6

If the stopping criteria are not met, the algorithm continues with parent selection, crossover, and mutation to generate a new population. This periodic process continues until the stopping criteria are satisfied, at which point the best feature set is selected. The final model, using these optimal features, is applied to the test data, which classifies the instances into COVID-19, Asthma, Pneumonia, or Normal categories. This bio-inspired GEP ensures robust hyperparameter tuning and enhances the classifier's accuracy.

Table 3. outlines hyperparameters for a DL model, specifying a population size of 50 and 10 generations for evolutionary optimization. The dropout rate is set at 0.5 to mitigate overfitting, while a batch size of 32 is used for training iterations. The learning rate is 0.001, which controls the step size during optimization. Additionally, a crossover rate of 0.6 is used for GA operations, and the fitness value represents the evaluation metric for the model's performance.

5. EXPERIMENTAL DETAILS

This section introduces the datasets, evaluation metrics, and training information for the experiments. Then we do broad comparison investigations with various models and loss functions, and lastly, analyze the outcomes.

5.1. Dataset Description

The dataset comprises Lung images graded into four classes namely COVID-19, Asthma, Pneumonia, and Normal. Images were collected from publicly available repositories [29-31].

Table 4. Dataset Description

S. N0	Class Name	No of Samples
1	Covid-19	1250
2	Pneumonia	1340
3	Normal	1545
4	Asthma	2250

Table 4. summarizes the dataset Sarscov-2 and Covid-CT used for classification, comprising 1,250 Covid-19 samples, 1,340 Pneumonia samples, 1,545 Normal samples, and 2,250 Asthma samples. The dataset shows a slight imbalance, with the most samples in the Asthma class and the fewest in the Covid-19 class. So, to make the class balance GANs are used in preprocessing. This combination not only improves data quality but also improves the overall efficiency of the GAN in distinguishing between the given classes. To enhance data, a data generator modifies existing images with transformations like rotation, while a GAN generates entirely new, realistic images from scratch. GANs provide richer and more diverse samples than basic data generators.

5.2. Hardware and Software

All experiments are executed on a Jupyter notebook using Python 3.8, running on hardware with a 2.3 GHz Intel Core i9 processor, 16 GB of 2400 MHz DDR4 RAM, and Intel UHD Graphics 630 with 1536 MB of memory.

5.3. Results & Discussion

In this study, the performance of three distinct pre-trained DL models, namely ResNet 50, VGG-16, Resnet101, and ChesxNet, was analyzed against the new EfficientNet+GEP model. To test classification performance, a number of metrics were utilized, including accuracy, sensitivity, specificity, positive predictive value, and negative predictive value (NPV). These measurements were essential for assessing discrimination's effectiveness. The false positive index (FP), true positive index (TP), true negative index (TN), and false negative index (FN) were the four operation indices that were computed [38-40]. An FP (False Positive) case indicates that COVID-19 was incorrectly identified as viral pneumonia, whereas a TP (True Positive) case indicates that COVID-19 was correctly identified. On the other hand, a TN (True Negative) case indicates that viral pneumonia has been accurately identified as COVID-19, whereas a FN (False Negative) instance occurs when COVID-19 is successfully identified as viral pneumonia.

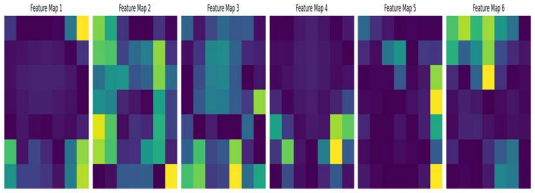

Figure 9. The visualization of Lung CT scans through feature maps

Figure 9. visualizes six feature maps from the EfficientNet model, extracted just before the dense layers. Each feature map represents a 2D grid. Each cell's color intensity and shade indicate the presence and strength of specific learned features within the input image, such as edges, textures, or patterns. The proposed method's classification performance is fully evaluated utilizing a variety of assessment metrics, including the F1 score, PRE (precision), ACC (accuracy), and recall. The F1 score combines accuracy and recall. Accuracy is the ratio of properly predicted samples to total predicted samples; precision is the percentage of all positive samples that are truly positive; and recall is the percentage of all positive samples that are correctly anticipated to be positive. Below are the formulas for these evaluation metrics.

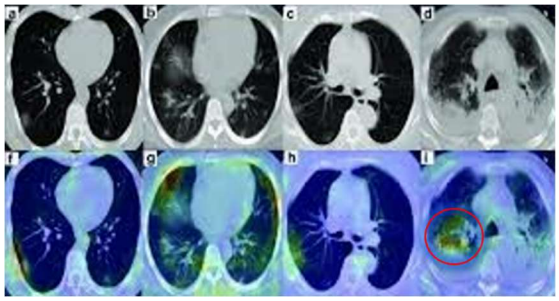

Figure 10. Heat Map of lung CT-scan for identified disease

For instance, the red-circled area in the last heatmap in Figure. 10 suggests the model's detection of significant features such as ground-glass opacities or consolidations, commonly associated with COVID-19 or Pneumonia. These visualizations indicate the model's interpretability and accuracy in identifying specific disease features. There are four scenarios of the estimated values of the model are investigated in the assessment of classification problems. Positive

classifications that are accurately and wrongly predicted are referred to as True Positives (TP) and False Positives (FP), respectively [41]. True Negatives (TN) and False Negatives (FN) indicate negatively predicted classes that are mistakenly and correctly predicted, respectively. These words are used to assess a model's performance by looking at its precision, sensitivity, and accuracy.

Accuracy
The most frequent statistic for assessing classification model success is the ratio of properly predicted samples to total samples.

$$Success\ rate = \frac{(TP+TN)}{(TP+FP+TN+FN)} \quad (12)$$

Precision
It is defined as the proportion of correctly predicted classes that are positive (TP) to all predicted positive cases (TP+FP), or the measure of precision (precision).

$$Precision = \frac{TP}{(TP+FP)} \quad (13)$$

The provided Figure 11. illustrates the training and validation loss for a model over 10 epochs before optimization. The training loss, illustrated by the red line, drops sharply from around 4.0 to less than 0.5 during the first few epochs, suggesting that the model is effectively learning from the training data. Similarly, the green line represents the validation loss, which reduces fast and stabilizes at around the same position as the training loss, indicating strong generalization without overfitting. The best epoch, marked by a blue dot at epoch 9, highlights the point of optimal performance on the validation set. To further evaluate the model's performance, accuracy and precision metrics for each epoch are essential.

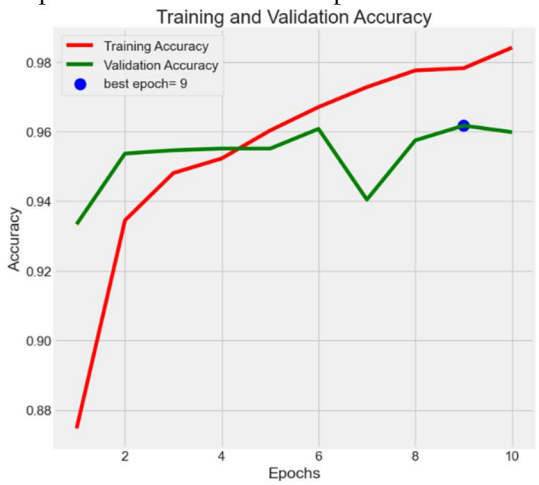

Figure 11. Training and Validation Accuracy

Figure 11. illustrates the training and validation accuracy of an EfficientNet model over 10 epochs. The training accuracy, represented by the red line, steadily increases, indicating that the model is effectively learning from the training data. Meanwhile, the validation accuracy, represented by the green line, fluctuates slightly, suggesting some variability in the model's performance on unseen data, but overall shows an improvement. The best performance on the validation set is marked at epoch 9 (indicated by the blue dot), where the accuracy peaks before slight fluctuations, potentially hinting at minor overfitting. This suggests that while the model is learning well, further optimization, such as implementing regularization techniques or using early stopping, could improve generalization.

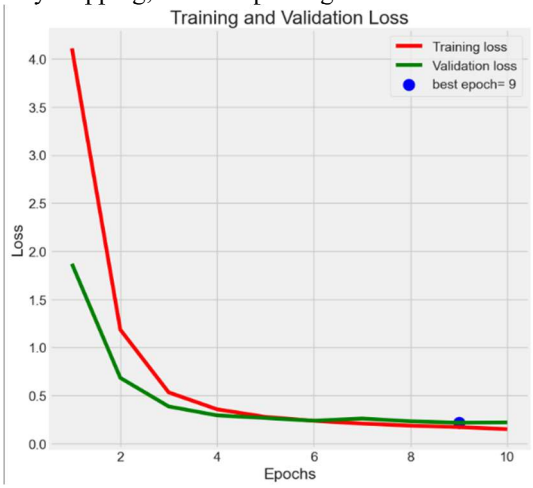

Figure 12. Training and Validation Loss of EfficientNet

Figure 12. shows training and validation loss over 10 epochs. Training loss (red line) decreases steadily, while validation loss (green line) also drops and stabilizes, indicating good model generalization. The best performance is at epoch 9 (blue dot), where both losses are lowest, suggesting optimal learning.

The confusion matrix in Figure.13 shows the classification results for four classes: Asthma, COVID-19, Normal, and Pneumonia. True positives are on the diagonal, indicating 356 correctly classified Asthma cases, 562 COVID-19 cases, 983 Normal cases, and 132 Pneumonia cases. Off-diagonal values represent misclassifications, with notable confusion between COVID-19 and Normal classes, suggesting the model needs improvement in distinguishing between these similar classes.

Table 5. shows the performance metrics of an EfficientNet model for a classification task, with an

accuracy of 85%, an F1-Score of 75%, and a precision of 72%. These results indicate moderate effectiveness in balancing true positives and minimizing false positives. To enhance these metrics, GEP approach is employed for hyperparameter tuning.

Figure 13. Confusion Matrix

Table 5. Results obtained by EfficientNet model

Metrics	Values
Accuracy	85
F1-Score	75
Precision	72

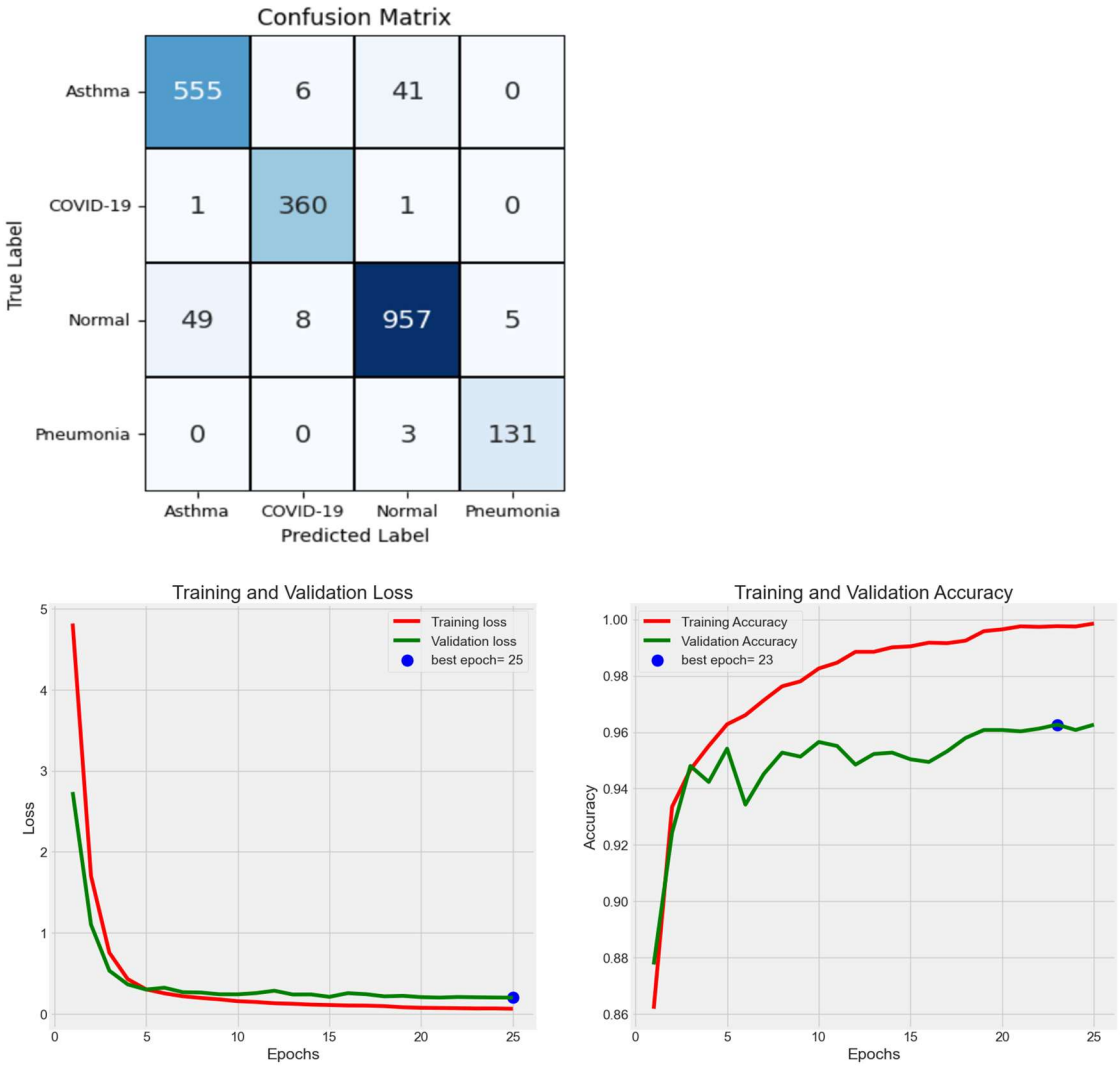

Figure 14. Results with EfficientNet with GEP

Figure 14. displays the training (red line) and validation (green line) loss over 25 epochs, with the lowest validation loss (blue marker) at epoch 25. Both losses drop sharply in the initial epochs, indicating rapid learning and error reduction by the model. Subsequent to this steep fall, both training and validation losses stabilize at low values, demonstrating the model's consistent performance on both datasets. The close alignment of the two lines suggests that overfitting is minimal, as the model maintains low error rates on unseen data, with the optimal performance observed at the final epoch.

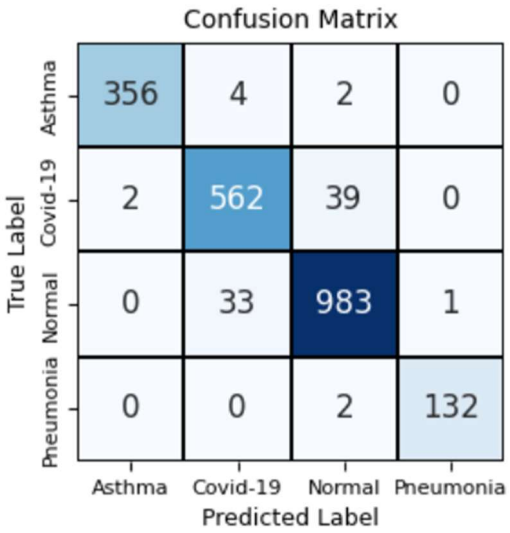

Figure 15. Confusion Matrix

It shows the importance of pre-processing in the GEP model. For the multi-class classification of healthy and various lung disease statuses, confusion matrices of EfficientNetB4+GEP, best performed pre-trained DL model are depicted in Figure. 15. The above Confusion Matrix indicates that the model accurately predicted most instances, with 356 correct predictions for asthma, 562 for COVID-19, 983 for normal, and 132 for pneumonia. There are minimal misclassifications, such as 4 asthma cases misclassified as COVID-19 and 33 normal cases misclassified as COVID-19.

6. ABLATION STUDY

To explore the effectiveness of the key components in the proposed EfficientNet+GEP structure, an ablation study is conducted and the results are shown in Table 6. As observations from Table. 6, the impact of incorporating a GEP into the EfficientNet model by comparing its performance against two other models: a baseline CNN and the standard EfficientNet B4. Ablation studies were conducted to assess how the proposed strategies influenced the model's performance. Specifically, it compares the impact of adding GEP (with and without) and investigates the performance differences between optimized EfficientNet with GEP and plain EfficientNet. Table 6 shows that each component greatly improves model accuracy across datasets, giving useful information about the usefulness of improved EfficientNet with GEP. Using EfficientNet, 2%-4% approximate increase in accuracy was observed and a 5%-11% increase in efficiency after optimizing it with GEP.

The table 6. compares the performance of two trained models: EfficientNet B4 and EfficientNet combined with GEP. EfficientNet B4 achieved an accuracy of 85.6%, a validation loss of 0.31, and an F1-score of 75%. In contrast, the model combining EfficientNet with GEP significantly outperformed EfficientNet B4, achieved an accuracy of 96.5%, a validation loss of 0.05, and an F1-score of 91%. This indicates that incorporating GA with EfficientNet improves the model's performance across all measured metrics

Table 6. Comparison and performances of the model with some existing techniques

S. No	Trained Model	Accuracy (%)	Validation loss	F1-Score (%)
1	SVM+ Dragonfly [42]	76	0.51	84
2	ResNet 101 V2 and CheXnet [43]	88	-	76
3	CNN-OELMnet [44]	85	-	91
4	VGG 19 and Resnet-50 [45]	87	0.42	80
5	EfficientNet B4 without GEP	85.6	0.31	75
6	EfficientNet B4+GEP	96.5	0.05	91

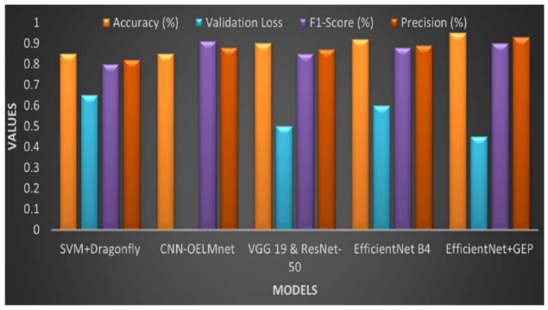

Figure 16. Visualization of Proposed Vs Existing

The Figure 16. illustrates the performance of three models (SVM+Dragonfly, EfficientNet B4, and EfficientNet+GEP) using Four metrics: accuracy, validation loss, F1-score and Precision [46]. The inexpensive EfficientNet with GEP model significantly decreases computing complexity and time requirements.

7. CONCLUSION

This study examined how the GEP and EfficientNet B4 can identify lung diseases such as COVID-19, normal conditions, pneumonia, and asthma. EfficientNet B4, noted for its high accuracy and efficient architecture, was used to extract features for this essential task. The GEP was used to optimize important hyperparameters like learning rates and dropout rates, improving model performance by efficiently navigating the hyperparameter space for optimal outcomes. The combination of EfficientNet B4 and GEP showed promising results in accurately classifying lung diseases. The model's architecture, including convolutional and MBConv blocks, effectively extracts features from medical imaging data. The use of Swish activation functions in the hidden layers contributed to improved training dynamics and model performance. GAP was used to compress the feature maps to a single value per channel, allowing for an efficient transition to the classification layer. The final Dense layer, which used a softmax activation function, guaranteed correct probabilistic classification of all four circumstances.

The combination of EfficientNet B4's depth and the optimization power of GEP provided a balanced commutation between computational efficiency and model accuracy. The aforementioned highlights the effectiveness of using advanced neural network architectures coupled with optimization techniques to tackle complex classification tasks in medical Visualization. Down the line, the dataset could be expanded to include diverse samples and incorporate multi-modal data. Additionally, exploring advanced neural architectures and real-time adaptation mechanisms could further improve model performance and clinical applicability.

CONFLICTS OF INTEREST: The authors declare no conflicts of interest to report regarding the present study.

DATA AVAILABILITY: The data and Code available at http://www.kaggle.com/plameneduardo/sarscov2-ctscan-dataset

AUTHOR CONTRIBUTIONS

Yellepeddi Samba Siva Krishna Assish contributed to the problem analysis and writing of the article. Kuppusamy P, as the supervisor, formulated the problem statement and provided guidance on organizing the manuscript, ensuring accurate interpretations. All authors have reviewed and approved the final version of the manuscript for submission.

ACKNOWLEDGMENTS

The authors express their gratitude to the editors and reviewers.

REFERENCES

[1]. Xie, Xingzhi, Zheng Zhong, Wei Zhao, Chao Zheng, Fei Wang, and Jun Liu. "Chest CT for typical coronavirus disease 2019 (COVID-19) pneumonia: relationship to negative RT-PCR testing." Radiology 296, no. 2 (2020): E41-E45.

[2]. Kanne, Jeffrey P., Brent P. Little, Jonathan H. Chung, Brett M. Elicker, and Loren H. Ketai. "Essentials for radiologists on COVID-19: an update—radiology scientific expert panel." Radiology 296, no. 2 (2020): E113-E114.

[3]. Tariq, Zeenat, Sayed Khushal Shah, and Yugyung Lee. "Lung disease classification using deep convolutional neural network." In 2019 IEEE international conference on bioinformatics and biomedicine (BIBM), pp. 732-735. IEEE, 2019.

[4]. Bansal, Shrey, Mukul Singh, R. K. Dubey, and Bijaya Ketan Panigrahi. "Multi-objective genetic algorithm based deep learning model for automated covid-19 detection using

medical image data." Journal of Medical and Biological Engineering 41, no. 5 (2021): 678-689.

[5]. Bhargava, Anuja, Atul Bansal, and Vishal Goyal. "Machine learning-based automatic detection of novel coronavirus (COVID-19) disease." Multimedia Tools and Applications 81, no. 10 (2022): 13731-13750.

[6]. Roser, Max, Hannah Ritchie, Esteban Ortiz-Ospina, and Joe Hasell. "Coronavirus disease (COVID-19)–Statistics and research." Our World in data 4 (2020): 1-45.

[7]. Su, Yuanjie, Guorui Chen, Chunxu Chen, Qichen Gong, Guangzhong Xie, Mingliang Yao, Huiling Tai, Yadong Jiang, and Jun Chen. "Self-powered respiration monitoring enabled by a triboelectric nanogenerator." Advanced Materials 33, no. 35 (2021): 2101262.

[8]. Loey, Mohamed, Shaker El-Sappagh, and Seyedali Mirjalili. "Bayesian-based optimized deep learning model to detect COVID-19 patients using chest X-ray image data." Computers in Biology and Medicine 142 (2022): 105213.

[9]. Reddy, A. Siva Krishna, KN Brahmaji Rao, Narasimha Reddy Soora, Kotte Shailaja, NC Santosh Kumar, Abel Sridharan, and J. Uthayakumar. "Multi-modal fusion of deep transfer learning based COVID-19 diagnosis and classification using chest x-ray images." Multimedia Tools and Applications 82, no. 8 (2023): 12653-12677.

[10]. Pasha, Akram, and P. H. Latha. "Bio-inspired dimensionality reduction for Parkinson's disease (PD) classification." Health information science and systems 8, no. 1 (2020): 13.

[11]. Ayyar, Meghna P., Jenny Benois-Pineau, and Akka Zemmari. "A hierarchical classification system for the detection of Covid-19 from chest X-ray images." In Proceedings of the IEEE/CVF international conference on computer vision, pp. 519-528. 2021.

[12]. Wang, Linda, Zhong Qiu Lin, and Alexander Wong. "Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images." Scientific reports 10, no. 1 (2020): 19549.

[13]. Qi, Xiao, David J. Foran, John L. Nosher, and Ilker Hacihaliloglu. "Multi-Scale Feature Fusion using Parallel-Attention Block for COVID-19 Chest X-ray Diagnosis." arXiv preprint arXiv:2304.12988 (2023).

[14]. Goyal, Shimpy, and Rajiv Singh. "Detection and classification of lung diseases for pneumonia and Covid-19 using machine and deep learning techniques." Journal of Ambient Intelligence and Humanized Computing 14, no. 4 (2023): 3239-3259.

[15]. Bougourzi, Fares, Riccardo Contino, Cosimo Distante, and Abdelmalik Taleb-Ahmed. "CNR-IEMN: A Deep Learning based approach to recognise COVID-19 from CT-scan." In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8568-8572. IEEE, 2021.

[16]. Shah, Vruddhi, Rinkal Keniya, Akanksha Shridharani, Manav Punjabi, Jainam Shah, and Ninad Mehendale. "Diagnosis of COVID-19 using CT scan images and deep learning techniques." Emergency radiology 28 (2021): 497-505.

[17]. Afshar, Parnian, Shahin Heidarian, Farnoosh Naderkhani, Moezedin Javad Rafiee, Anastasia Oikonomou, Konstantinos N. Plataniotis, and Arash Mohammadi. "Hybrid deep learning model for diagnosis of COVID-19 using CT scans and clinical/demographic data." In 2021 IEEE International Conference on Image Processing (ICIP), pp. 180-184. IEEE, 2021.

[18]. Demir, Fatih, Kürşat Demir, and Abdulkadir Şengür. "DeepCov19Net: automated COVID-19 disease detection with a robust and effective technique deep learning approach." New Generation Computing 40, no. 4 (2022): 1053-1075.

[19]. Chouhan, Vikash, Sanjay Kumar Singh, Aditya Khamparia, Deepak Gupta, Prayag Tiwari, Catarina Moreira, Robertas Damaševičius, and Victor Hugo C. De Albuquerque. "A novel transfer learning based approach for pneumonia detection in chest X-ray images." Applied Sciences 10, no. 2 (2020): 559.

[20]. Nayak, Soumya Ranjan, Deepak Ranjan Nayak, Utkarsh Sinha, Vaibhav Arora, and Ram Bilas Pachori. "Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: A comprehensive study." Biomedical Signal Processing and Control 64 (2021): 102365.

[21]. Yang, Hang, Liyang Wang, Yitian Xu, and Xuhua Liu. "CovidViT: a novel neural network with self-attention mechanism to detect Covid-19 through X-ray images." International

journal of machine learning and cybernetics 14, no. 3 (2023): 973-987.

[22]. Godbin, A. Beena, and S. Graceline Jasmine. "Screening of COVID-19 based on GLCM features from CT images using machine learning classifiers." SN Computer Science 4, no. 2 (2022): 133.

[23]. Tiwari, Ravi Shekhar, Tapan Kumar Das, Kathiravan Srinivasan, and Chuan-Yu Chang. "Conceptualising a channel-based overlapping CNN tower architecture for COVID-19 identification from CT-scan images." Scientific Reports 12, no. 1 (2022): 18197.

[24]. Hassan, Esraa, Mahmoud Y. Shams, Noha A. Hikal, and Samir Elmougy. "Detecting COVID-19 in chest CT images based on several pre-trained models." Multimedia Tools and Applications (2024): 1-21.

[25]. Hermawati, Fajar Astuti, Bambang Riyanto Trilaksono, Anto Satriyo Nugroho, Elly Matul Imah, Telly Kamelia, Tati LER Mengko, Astri Handayani et al. "Detection method of viral pneumonia imaging features based on CT scan images in COVID-19 case study." MethodsX 12 (2024): 102507.

[26]. Mittal, Vasu, and Akhil Kumar. "COVINet: A hybrid model for classification of COVID and non-COVID pneumonia in CT and X-Ray imagery." International Journal of Cognitive Computing in Engineering 4 (2023): 149-159.

[27]. Siddiqua, Morseda, and Zannatul Ferdousee. "Deep Transfer Learning Based Approaches for Classification of Chest CT Scans: Normal, COVID-19, and Pneumonia." In 2024 6th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT), pp. 841-846. IEEE, 2024.

[28]. Hossain, Md Sabbir, Md Farukuzzaman Faruk, Azmain Yakin Srizon, SM Mahedy Hasan, Md Shariful Chowdhury, Md Rakib Hossain, Md Nazmul Islam, and Md Faruk Hossain. "A Customized 3D CNN Integrated with Convolutional Block Attention Module for Precise Diagnosis of COVID-19 and Pneumonia from CT Scan Images." In 2023 26th International Conference on Computer and Information Technology (ICCIT), pp. 1-6. IEEE, 2023.

[29]. Zhao, Aite, Huimin Wu, Ming Chen, and Nana Wang. "A multi-level feature attention network for COVID-19 detection based on multi-source medical images." Multimedia Tools and Applications (2024): 1-32.

[30]. Gao, Zilin, Jiangtao Xie, Qilong Wang, and Peihua Li. "Global second-order pooling convolutional networks." In Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition, pp. 3024-3033. 2019.

[31]. Rafi, Taki Hasan, Raed M. Shubair, Faisal Farhan, Md Ziaul Hoque, and Farhan Mohd Quayyum. "Recent advances in computer-aided medical diagnosis using machine learning algorithms with optimization techniques." IEEE Access 9 (2021): 137847-137868.

[32]. Gifani, Parisa, Ahmad Shalbaf, and Majid Vafaeezadeh. "Automated detection of COVID-19 using ensemble of transfer learning with deep convolutional neural network based on CT scans." International journal of computer assisted radiology and surgery 16 (2021): 115-123.

[33]. Ju, Hong, Yanyan Cui, Qiaosen Su, Liran Juan, and Balachandran Manavalan. "CODE-NET: A deep learning model for COVID-19 detection." Computers in Biology and Medicine (2024): 108229.

[34]. Yang, Xingyi, Xuehai He, Jinyu Zhao, Yichen Zhang, Shanghang Zhang, and Pengtao Xie. "COVID-CT-dataset: a CT scan dataset about COVID-19." arXiv preprint arXiv:2003.13865 (2020).

[35]. Angelov, Plamen, and Eduardo Soares. "Explainable-by-design approach for covid-19 classification via ct-scan." (2020).

[36]. Zhang, Mengting, and Xiuxia Tian. "Transformer architecture based on mutual attention for image-anomaly detection." Virtual Reality & Intelligent Hardware 5, no. 1 (2023): 57-67

[37]. Kulkarni, Ajay, Deri Chong, and Feras A. Batarseh. "Foundations of data imbalance and solutions for a data democracy." In Data democracy, pp. 83-106. Academic Press, 2020.

[38]. Raji, Ismail Damilola, Habeeb Bello-Salau, Ime Jarlath Umoh, Adeiza James Onumanyi, Mutiu Adesina Adegboye, and Ahmed Tijani Salawudeen. "Simple deterministic selection-based genetic algorithm for hyperparameter tuning of machine learning models." Applied Sciences 12, no. 3 (2022): 1186.

[39]. Alibrahim, Hussain, and Simone A. Ludwig. "Hyperparameter optimization: Comparing genetic algorithm against grid search and bayesian optimization." In 2021 IEEE Congress on Evolutionary Computation (CEC), pp. 1551-1559. IEEE, 2021.

[40]. Sen, Shibaprasad, Soumyajit Saha, Somnath Chatterjee, Seyedali Mirjalili, and Ram Sarkar. "A bi-stage feature selection approach for COVID-19 prediction using chest CT images." Applied Intelligence 51 (2021): 8985-9000.

[41]. Pradhan, Kanchan, Priyanka Chawla, and Sanyog Rawat. "A deep learning-based approach for detection of lung cancer using self adaptive sea lion optimization algorithm (SA-SLnO)." Journal of Ambient Intelligence and Humanized Computing 14, no. 9 (2023): 12933-12947.

[42]. Liu, Weili, Bo Wang, Yucheng Song, and Zhifang Liao. "Radiological image analysis using effective channel extension and fusion network based on COVID CT images." Journal of Radiation Research and Applied Sciences 17, no. 3 (2024): 100965.

[43]. Visuña, Lara, Dandi Yang, Javier Garcia-Blas, and Jesus Carretero. "Computer-aided diagnostic for classifying chest X-ray images using deep ensemble learning." BMC Medical Imaging 22, no. 1 (2022): 178.

[44]. Agarwal, Saurabh, K. V. Arya, and Yogesh Kumar Meena. "CNN-O-ELMNet: Optimized Lightweight and Generalized Model for Lung Disease Classification and Severity Assessment." IEEE Transactions on Medical Imaging (2024).

[45]. Farjana, Afia, Fatema Tabassum Liza, Miraz Al Mamun, Madhab Chandra Das, and Md Maruf Hasan. "SARS CovidAID: Automatic detection of SARS CoV-19 cases from CT scan images with pretrained transfer learning model (VGG19, RESNet50 and DenseNet169) architecture." In 2023 International Conference on Smart Applications, Communications and Networking (SmartNets), pp. 1-6. IEEE, 2023.

[46]. Yellepeddi, Samba Siva Krishna Assish, and P. Kuppusamy. "An Automatic Detection and Severity Levels of COVID-19 Using Convolutional Neural Network Models." In Data Science in the Medical Field, edited by Seifedine Kadry and Shubham Mahajan, 15-24. Academic Press, 2025. https://doi.org/10.1016/B978-0-443-24028-7.00003-9.