



OPEN

An augmentation aided concise CNN based architecture for COVID-19 diagnosis in real time

Balraj Preet Kaur¹, Harpreet Singh², Rahul Hans¹, Sanjeev Kumar Sharma³, Chetna Kaushal⁴, Md. Mehedi Hassan⁵ & Mohd Asif Shah^{6,7,8}✉

Over 6.5 million people around the world have lost their lives due to the highly contagious COVID 19 virus. The virus increases the danger of fatal health effects by damaging the lungs severely. The only method to reduce mortality and contain the spread of this disease is by promptly detecting it. Recently, deep learning has become one of the most prominent approaches to CAD, helping surgeons make more informed decisions. But deep learning models are computation hungry and devices with TPUs and GPUs are needed to run these models. The current focus of machine learning research is on developing models that can be deployed on mobile and edge devices. To this end, this research aims to develop a concise convolutional neural network-based computer-aided diagnostic system for detecting the COVID 19 virus in X-ray images, which may be deployed on devices with limited processing resources, such as mobile phones and tablets. The proposed architecture aspires to use the image enhancement in first phase and data augmentation in the second phase for image pre-processing, additionally hyperparameters are also optimized to obtain the optimal parameter settings in the third phase that provide the best results. The experimental analysis has provided empirical evidence of the impact of image enhancement, data augmentation, and hyperparameter tuning on the proposed convolutional neural network model, which increased accuracy from 94 to 98%. Results from the evaluation show that the suggested method gives an accuracy of 98%, which is better than popular transfer learning models like Xception, Resnet50, and Inception.

Coronavirus¹ was identified in Wuhan, China, in 2019, and it has affected more than 760 million people around the globe². The virus causes respiratory diseases such as Middle East Respiratory Syndrome, Severe Acute Respiratory Syndrome (SARS)³, and other deadly complications. The most common symptoms are cough, sore throat, headache, fever and fatigue (<https://covid19.who.int/>). The virus is passed from person to person by droplets of breath. During past COVID 19 waves, the sudden surge in cases made it difficult for the laboratories to confirm positive or negative cases using RT-PCR as it is a time-consuming method and has high false-negative rates⁴, and is costly also. Therefore, development of real time diagnostic tools, which can be executed in mobile and edge devices is the need of the hour⁵. Since most diagnostic centers already have X-ray machines, and because acquiring an X-ray takes less time than getting the RT-PCR done, using chest X-rays of patients⁶ satisfies the urgent need for a speedy diagnostic approach.

Deep learning^{7,8} is one of the most promising techniques that provides efficient results in the accurate diagnosis of the diseases from images and is widely used in the medical field to diagnose severe diseases at early stages⁹. It is made up of input layer, activation functions, hidden layers and also output layer. The mathematical equation in each step with feed forward and backward functions can help in finding better results¹⁰. An activation function is used to activate and deactivate the neurons and basically defines the output of a node. Convolutional neural networks (CNN)^{11,12} are deep learning neural network made up of neurons which are experienced, self-optimized and are used primarily by researchers working in the field of disease diagnosis from images. CNN's key popularity

¹Department of Computer Science and Engineering, DAV University, Jalandhar, India. ²Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, India. ³Department of Computer Science and Applications, DAV University, Jalandhar, India. ⁴Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab 140401, India. ⁵Computer Science and Engineering Discipline, Khulna University, Khulna 9208, Bangladesh. ⁶Department of Economics, Kebri Dehar University, Kebri Dehar, 250, Ethiopia. ⁷Centre of Research Impact and Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, 140401, Punjab, India. ⁸Division of Research and Development, Lovely Professional University, Phagwara, 144001, Punjab, India. ✉email: drmohdasifshah@kdu.edu.et

is attributed to its ability to automatically learn functions from domain-specific images¹³. Furthermore, transfer learning models¹⁴ saves knowledge from one problem and can apply that knowledge on another problem. But the conventional CNN models such as Resnet50, AlexNet, Inception and Xception, etc., cannot be run on low computing power devices such as tablets, embedded chips, mobile phones and hence cannot be given to real time applications^{15,16}. These conventional models are also complex, need a lot of training time. To overcome these shortcomings, the lightweight¹⁷ and concise models of CNNs are being developed which are having lesser number of parameters than the conventional CNNs so that they can be executed on devices with low computing power and smaller memory requirements. Figure 1 shows the features of concise CNNs.

In the case of CNNs, there is also a need to preprocess the image to get a better classification and for that purpose, image enhancement techniques¹⁸ are used in which mask and filters upgrade the quality of image. In addition to that data augmentation techniques¹⁹ increase the training data to upgrade the successful rate of model. The goal of this study is to present a new, simple CNN-based model for diagnosing COVID 19 from X-ray pictures, and the proposed model has been compared to existing transfer learning methods using a number of different criteria. The following are the primary contributions of this research work: (1) a novel set of layers, as well as image enhancement and hyperparameter tuning of parameters, have been suggested for the classification of COVID-19, normal, and pneumonia cases. (2) In order to prevent the models from overfitting, data augmentation has also been carried out. (3) The proposed framework may be used as one of the effective methods for classifying data in the medical industry. Furthermore, it helps radiologists diagnose and treat ailments earlier.

The paper is divided into the sections listed below. The second section discusses the literature review and the third section explains our suggested model. Materials and methods are discussed in fourth section. Fifth section includes a description of the experimental outcomes. Conclusions and future work are discussed in sixth section.

Literature survey

Recently, most frequently research has been going on in the domain of diagnosis disease using CNN from images. This section summarizes some of the existing works for disease diagnosis. The comparison of models performances are shown in Table 1.

Litjens et al.²⁹ proposed application aspects in deep learning. The different deep learning techniques extracted the spatial features from sophisticated image data i.e. CT, X-ray images, color Fundus images, ultrasound image

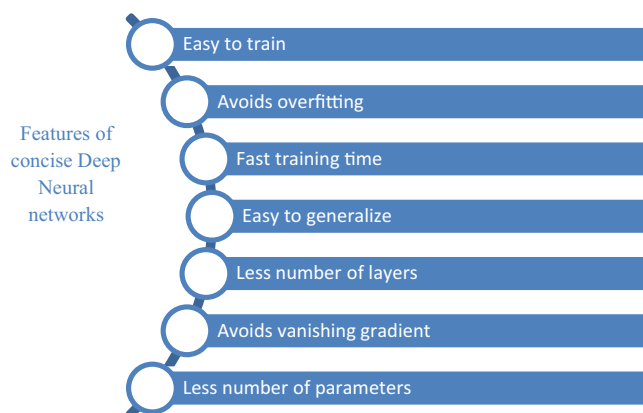


Figure 1. Features of concise CNNs.

Author's (year)	Disease	Image type	Models used	Accuracy (%)
Anubhav Sharma et al. (2022) ²⁰	COVID-19	Chest X-ray	CNN Mobile Net, VGG16, Resnet50, Xception	96.4
Khalid EL Asnaoui et al. (2021) ²¹	COVID-19	Chest X-ray and CT dataset	CNN Mobile Net, VGG16, Resnet50, Xception	96
SadmanSakib et al. (2020) ²²	COVID-19	Chest X-ray	CNN, GAN, LSTM	93
MortezaHeidari et al. (2020) ¹⁹	COVID-19	Chest X-ray	CNN, VGG16, VGG19, Xception, Inception, DenseNet121	94.5
Aras M. Ismael et al. (2021) ²³	COVID-19	Chest X-ray	CNN, SVM	94.7
Naveena et al. (2022) ²⁴	Diabetes	PIMA and UCI	CNN, RNN and Crow search algorithm	96
Aggarwal et al. (2021) ²⁵	COVID-19	Chest X-ray	MobileNetV2, ResNet50, InceptionV3, NASNetMobile, VGG16, Xception, InceptionResNetV2 DenseNet121	97
Dipayan Das et al. (2020) ²⁶	COVID-19	Chest X-ray	Truncated Inception Net, Inception Net V3	99.6
JieHou et al. (2021) ²⁷	COVID-19	Chest X-ray	Deep convolution neural network (DCNN)	96
Azher Uddin et al. (2021) ²¹	COVID-19	Chest X-ray	VGG16, InceptionV3, MobileNetV2, and ResNet50	98
Shashwat Sanket et al. (2020) ²⁸	COVID-19	Chest X-ray	VGG16, InceptionV3, VGG19, and ResNet50	98.4

Table 1. State of the art techniques.

and implemented models, which can be helpful in hospitals to detect severe diseases such as diabetic retinopathy, skin lesion, bone fracture and breast cancer at their early stages. Kermany et al.³⁰ used optical coherence tomography images dataset to detect viral pneumonia and macular degeneration and diabetic retinopathy. Cao et al.³ introduced deep learning, and the image analysis is done by deep learning architecture such as RNN, CNN and Stacked machine auto encoder. With these models, the detection of pediatric pneumonia with chest X-ray images can be done. The authors also presented the challenges in handling unlabeled data, privacy issues in the medical field, and many more. Jaiswal et al.³¹ in their work used the region of interest, align convolution layer and pixel-wise segmentation of disease.

Toğaçar et al.¹⁴ proposed a minimum redundancy maximum relevance (mRMR) model for the diagnosis of pneumonia. The three knowledge transfer models, namely, VGG-16, Alex Net, and VGG-19, are used in the proposed architecture. Moreover, decision tree, linear Discriminant analysis, k-nearest neighbor, and support vector machines are used for grouping using features generated by transfer learning model. Singh et al.³² proposed multi-objective differential evolution model for the classification of the COVID 19 disease. An exponential crossover algorithm is used. The proposed model gives high accuracy as compared to adaptive neuron fuzzy inference system, artificial neural network, and CNN types.

Das et al.³³ designed an Xception model to diagnosed COVID infection using chest X-ray dataset containing three classes pneumonia and COVID 19 negative, COVID 19 positive, and other infections except for COVID. The features are extracted by using different masks applied to the convolution layer. As a loss function, the cross-entropy is utilized. Brunese et al.³⁴ built two models: the first model assesses whether the picture belongs to a healthy patient or a patient suffering from general pulmonary illness. If the patient has a general pulmonary condition, the X-ray picture is sent to the second model, which checks whether it is a COVID 19 patient or pulmonary disease only.

Liu et al.⁷ suggested a model for dental disease diagnosis utilizing a mask region-based convolution neural network with classification of seven different dental diseases. The model uses an IoT platform for patients to upload their dental images. A broad-level prototype is also given in the paper for dental image acquisition. Jain et al.³⁵ presented four phases of model where ResNet50 network is used to differentiate between bacterial pneumonia and pneumonia. Varela et al.³⁶ suggested approach uses feature extraction to minimize the number of pixels, grey level co-occurrence matrix features that focus on the spatial relationship between pixels, and the local binary patterns method to encode the pixel values. Marques et al.³⁷ has been suggested efficientnetb4 model is a convolution neural network. Ezzat et al.³⁸ suggested a technique to identify the optimal settings for hyperparameters, the gravitational search method is utilized as an optimization tool. The new method is contrasted with Social Ski Driver-Dennsenet121. Data preparation, hyperparameter selection, and the learning step for COVID 19 diagnosis are all part of the technique. Hassantabar et al.³⁹ has been proposed technique for detecting COVID 19 patients. Two approaches are utilized for diagnosis. The first is a deep neural network, while the second is an image segmentation approach for detecting diseased areas. Table 1 summarizes studies relating to COVID 19 and CNN architecture on chest x ray images (Type1) and PIMA and UCI (Type2), as well as information on additional approaches utilized in the papers.

Various studies have explored the application of deep learning techniques across diverse imaging modalities, including CT scans, X-rays, color fundus images, ultrasound, and optical coherence tomography^{40–42}. The investigated diseases range from diabetic retinopathy and skin lesions to bone fractures, breast cancer, viral pneumonia, and COVID-19^{13,43}. These studies employ a variety of deep learning architectures such as RNNs, CNNs, and stacked machine auto encoders. Notably, researchers have addressed challenges like handling unlabeled data and privacy issues in the medical field⁴⁴. Key findings include the efficacy of models like Xception for diagnosing COVID-19 from chest X-ray images, the use of multi-objective models for COVID-19 classification, and innovative approaches like dental disease diagnosis using a mask region-based CNN⁴⁵. The comparison in Table 1 underscores the performance of different models in COVID-19 diagnosis and CNN architecture across chest X-ray images and datasets like PIMA and UCI. Overall, these studies demonstrate the versatility and potential impact of deep learning in advancing early disease detection and diagnostic accuracy in medical imaging.

Proposed concise CNN based architecture

The framework of a convolutional neural network depends on the number of layers, activation function, optimizer, number of filters and batch size^{46–48}. Figure 2 represents the proposed architecture of the COVID 19 diagnosis structure. The proposed model has been derived from the baseline Efficient Net model⁴⁶. In contrast to the more complex architectures, the goal is to develop a concise CNN model that can identify picture modification⁴⁰. The layers of the efficient Net model have are modified by replacing the MBconv layer with a Conv2D layer and also by updating the values of filter in layers additionally, the dropout layer is added with a 0.4 value to reduce overfitting of the model and add regularization. The proposed architecture is a sequential model. Additionally, the layers are added in the sequence order to build the CNN architecture. The proposed CNN model contains Conv2D, Maxpooling2D, Dropout, the Relu activation function, dense/fully connected layer. The suggested model has nine total layers: three convolutional, three maxpooling, three Relu, two dense, one dropout, and a fully connected layer.

- The size of image as input is $224 \times 224 \times 3$, i.e., 224 is height and width of image and 3 is image channel value as RGB. The first convolution layer (L1) represent the first layer of model takes an input of size $224 \times 224 \times 3$ and has kernel size 3×3 which produces 32 features maps as result.
- The second convolution layer (L2) has 32 filters and has kernel size 3×3 which produces 32 features maps as result.
- The third convolution layer (L3) contain 64 filters with kernel size 3×3 which produces tensor of 64 features maps as result.

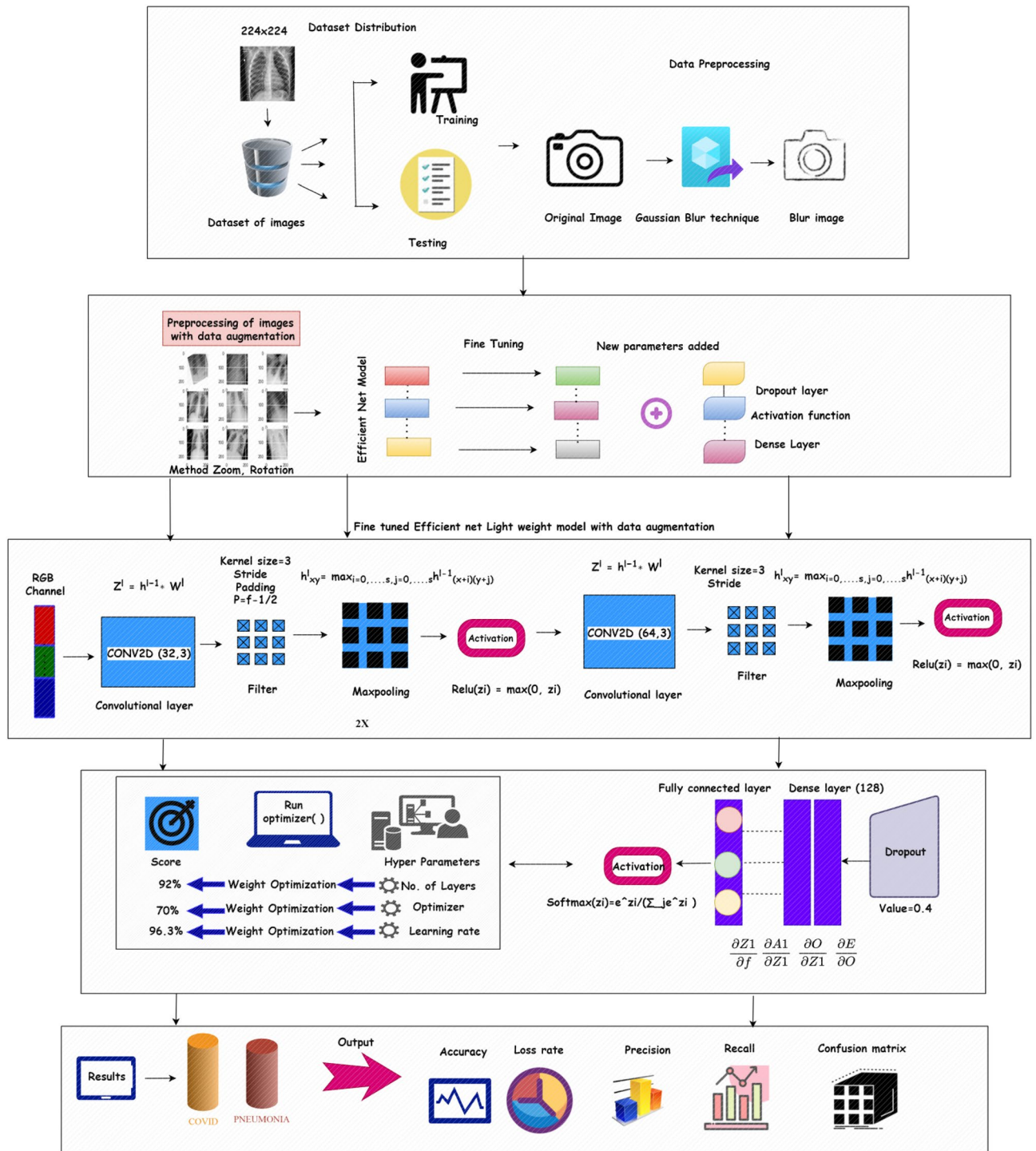


Figure 2. Architecture of proposed model (source of clipart images: draw.io).

- d. To tackle the overfitting problem, the above layers are followed by dropout layer with 64 filters having kernel size of 3×3 . Dropout layer is followed by flattening layer. The flatten layer converts the data into 1-D form.
- e. In last the dense/fully connected layer is added with 128 filters and the efficiency of the model is improved by Relu as activation function which produces 258 features. This layer produces the output.

Convolutional layers are used in conjunction with the most common Rectified linear unit (ReLU) activation function to increase the performance and generalization by introducing non-linearities to the network. The vanishing gradient issue that may be seen in other forms of activation functions is eliminated by ReLU by correcting the values of inputs less than zero. ReLU's key benefit is quicker execution, which shortens computation time. The maxpool2D is used with each convolution layer to extract the best features. The description of each

layer is represented in Table 2 in which Conv2D as T1 layer, Max_pooling2D as T2 layer, Dropout as T3 layer, Flatten as T4 layer and Fully connected as T5 layer.

The filter applied on the image is represented in Eq. (1). The h is a kernel and input image is represented as f . The resulting matrix of indexes of rows and columns is marked as q, r . Σ sign is used to add all values with limits j and k .

$$R[q, r] = [f * h][q, r] = \sum_j \sum_k h[j, k] f[q - j, r - k] \quad (1)$$

After this process, the filter is placed over the image and the value is multiplied by the value from the image. Then all values sum up and the feature map is generated. The padding is added to the image to fix the size in proper form. Equation (2) is used for padding.

$$p = \frac{f - 1}{2} \quad (2)$$

The preprocessing part with CONV2D now moves to the pooling process. The formula for the pooling function is defined in Eq. (3). In the pooling process, we find the maximum and average according to the pool type. It is a technique to get sample feature maps from all features. It extracts features that contain high value during the sliding window extraction process. Here, s is the stride, n_H is a size of height, nc is number of channels, n_w is size of width.

$$\text{Pooling} = \left(\left[\frac{n_H + 2p - f}{s} + 1 \right], \left[\frac{n_w + 2p - f}{s} + 1 \right], nc, \right) : s > 0, \quad (3)$$

For improved results, Relu is employed as an activation function in each CONV2D layer and maxpooling layer. The function work as Eq. (4), where x is the input value

$$\text{Relu}(x) = \max(0, x) \quad (4)$$

The convolution layer with maxpooling value is then direct to the feed forward function to calculate the value for the next step, Eq. (5) represents the functioning of this process

$$Z[I] = W[I] \cdot AF[I - 1] + b[I] \quad (5)$$

$$AF[I] = g[Z[I]] \quad (6)$$

Here, g is the activation function in Eq. (6), firstly the value of Z is calculated from the previous layer with W tensor and then bias b is added to it.

After calculating it, we move to the calculation of derivatives which will be used to update the value of parameter also known as gradient descent. The formula of a partial derivative as

$$dAF[I] = \frac{\partial L}{\partial A[I]} dZ[I] = \frac{\partial L}{\partial Z[I]} dW[I] = \frac{\partial L}{\partial W[I]} db[I] = \frac{\partial L}{\partial b[I]} \quad (7)$$

dW and db are parameters that work on the present layer. According to the chain rule, the result is Eq. (8)

$$dZ[I] = dAF[I] * g'(Z[I]) \quad (8)$$

After the backpropagation process, hyperparameters tuning is the next step which includes checking parameter values with different patterns based on the performance of the model. The parameter used for tuning are: loss function, learning rate, optimizer and number of neurons.

Layer no	Operation	No. of filter	Kernel	No. of parameter
1	Conv2D	32	3 × 3	896
2	Max_Pooling2D	32	1 × 1	N/A
3	Conv2D	32	3 × 3	9248
4	Max_Pooling2D	32	3 × 3	N/A
5	Conv2D	64	3 × 3	18,496
6	Max_Pooling2D	64	1 × 1	N/A
7	Dropout	64	3 × 3	N/A
8	Flatten	50,176	1 × 1	N/A
9	Fully connected layer	128	3 × 3	6,422,656

Table 2. Layer architecture of proposed model.

- (a) Loss function: It is used to compute the model error. The gradients may be calculated from the loss function and used to update the weights. To generate output, the suggested model uses a sparse categorical cross entropy loss function and the mathematical operation of which is shown in Eq. (9)

$$L = \sum_{j=1}^M y \log(\hat{y}_j) \quad (9)$$

where \hat{y} represent the outcome produced by the model and y represents the expected outcome.

- (b) Optimizer: The goal of an optimizer is to minimize losses by adjusting relevant model parameters like the learning rate and the weights. In the proposed approach, RMSprop is used as optimizer. The RMSprop takes the cumulative sum of the squared gradient represented in Eqs. (10) and (11)

$$wt + 1 = wt - \frac{\alpha t}{(vt + e)^{1/2}} * \left[\frac{\delta l}{\delta wt} \right] \quad (10)$$

$$vt = \beta vt - 1 + (1 - \beta) * \left[\frac{\delta l}{\delta wt} \right]^2 \quad (11)$$

Here, αt learning rate at time t , δwt derivative of weight at time t and δl derivative of the loss function, v_t sum of the square of past gradient, e small positive constant (10^{-8}) and β is moving average parameter (constant, 0.9). Dense Layer receives input from all neurons of the previous layer along with the Relu activation function. The dense layer return output is represented in Eq. (12)

$$o = g(\text{dot}(I, K) + b) \quad (12)$$

In the above equation, o is output g is the activation function, dot represents numpy function for calculation, I is input. K represents the weight data, while b is the bias used to optimize the model. Figure 3 depicts the study's step-by-step process. The classification process of CNNs is to process input images through convolutional, activation, pooling, and fully connected layers. Training involves optimizing weights via backpropagation to minimize a loss function. The trained model predicts image classes by analyzing learned features.

Materials and methods

The description of the dataset⁴⁹ that was utilized in the experiment is configured into two types. The firstly used dataset in the experiment is dataset with data augmentation⁴⁴. The second dataset used in the experiment is with image enhancement using hyper parameter tuning, data augmentation and Gaussian Blur.

Dataset description

The suggested model for detecting COVID 19 illness was evaluated using a dataset of publically accessible conventional chest X ray images⁵⁰. The collection includes 3616 COVID 19 positive cases, 10,192 Normal pictures, 6012 Lung Opacity, and 1345 viral Pneumonia images. Only two of the four types of images presented were taken into account in our experiments, i.e., COVID 19 positive and viral Pneumonia. Every image is a grayscale image consisting of 299×299 pixels. Figure 4 shows the sample image from each class of the test dataset. Total of 1000 images are taken from a dataset and then divided into different three samples.

Image enhancement

Gaussian blur feature is derived by blurring an image using Gaussian function. This technique upgrade the quality of an image and is helpful in finding inadequate information for image interpretation⁵¹. The spatial filtering, slicing, stretch, edge sharpening and other methods are used in this technique. The method reduces noise and increases smoothening of image. The process is achieved by convolving on image with Gaussian kernel. The formula used for the process is shown in Eq. (13)¹⁸:

$$G_{2D}(x, y, \sigma^2) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (13)$$

Here the distribution by standard deviation is denoted by σ and x, y are location indices. The Gaussian distribution mean value depends on the value of σ which influence the extent of blurring affect around pixel. The COVID and viral pneumonia images after and before Gaussian blur is shown in Figs. 5 and 6.

The `opencv2` is used to implement Gaussian blur. The three functions are used as argument in the process i.e. `img` used to modified the image, `sigma` used in the x and y direction and `truncate` used to determining the limits of the approx. The Gaussian filter takes the x, y pixel and returns a single number by calculating the weighted average based on the normal distribution.

Figure 5 shows the images of chest X ray of covid class with and without image enhancement using Gaussian Blur¹⁸ technique. Figure 6 shows the images of chest X-ray of Viral-pneumonia classes with and without image enhancement using Gaussian Blur technique. The paper results contain experimentation on two type of dataset i.e. with Gaussian Blur images dataset and without Gaussian Blur images dataset.

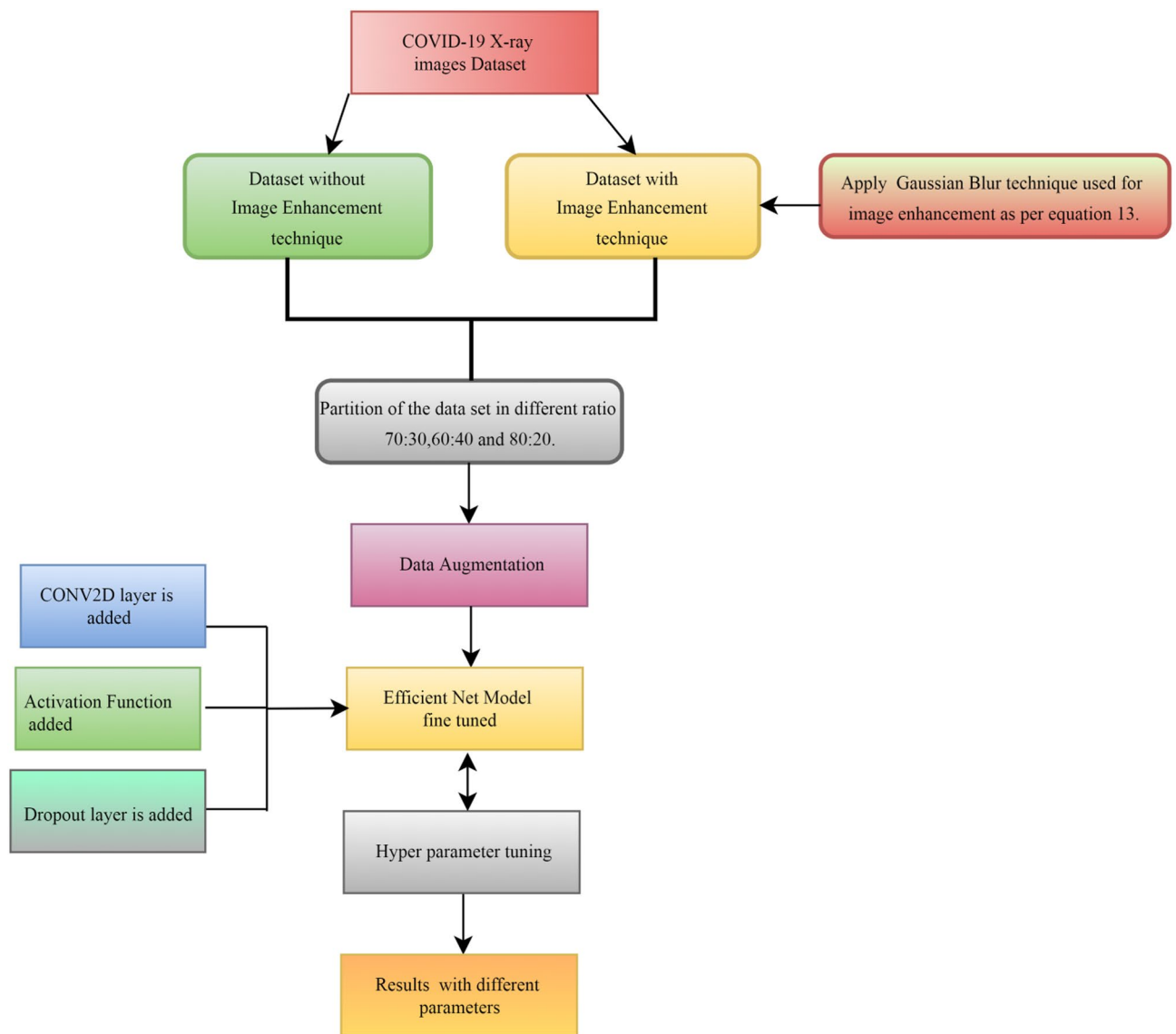


Figure 3. Phases undertaken in the proposed framework.

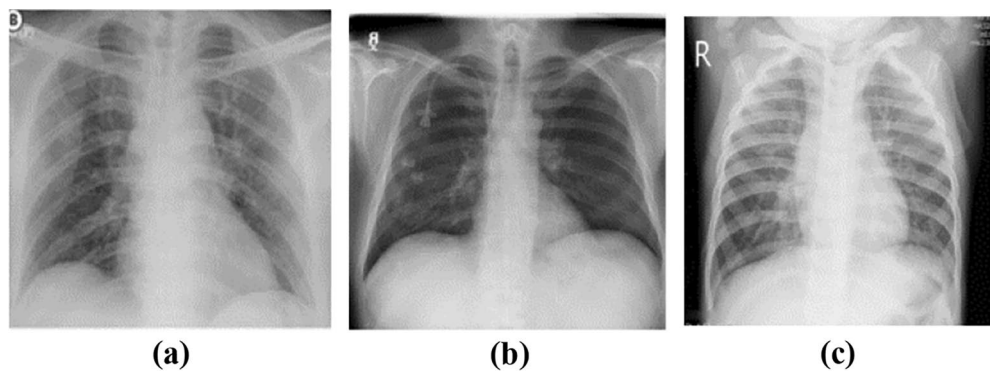


Figure 4. Sample image from each class (a) COVID 19 infected, (b) normal and (c) pneumonia.

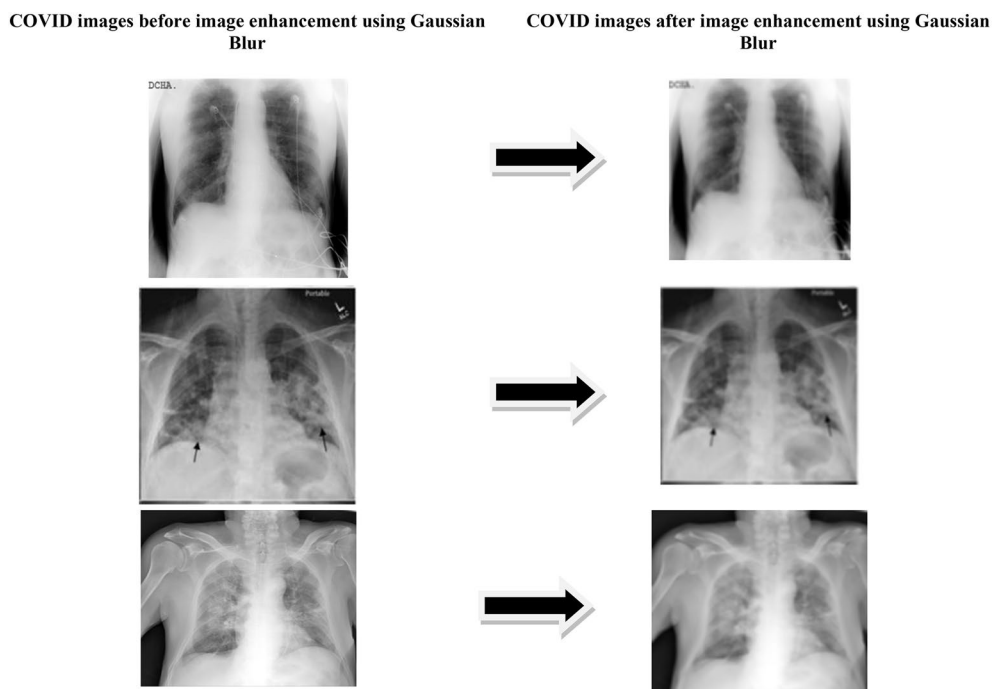


Figure 5. Image enhancement result on COVID dataset.

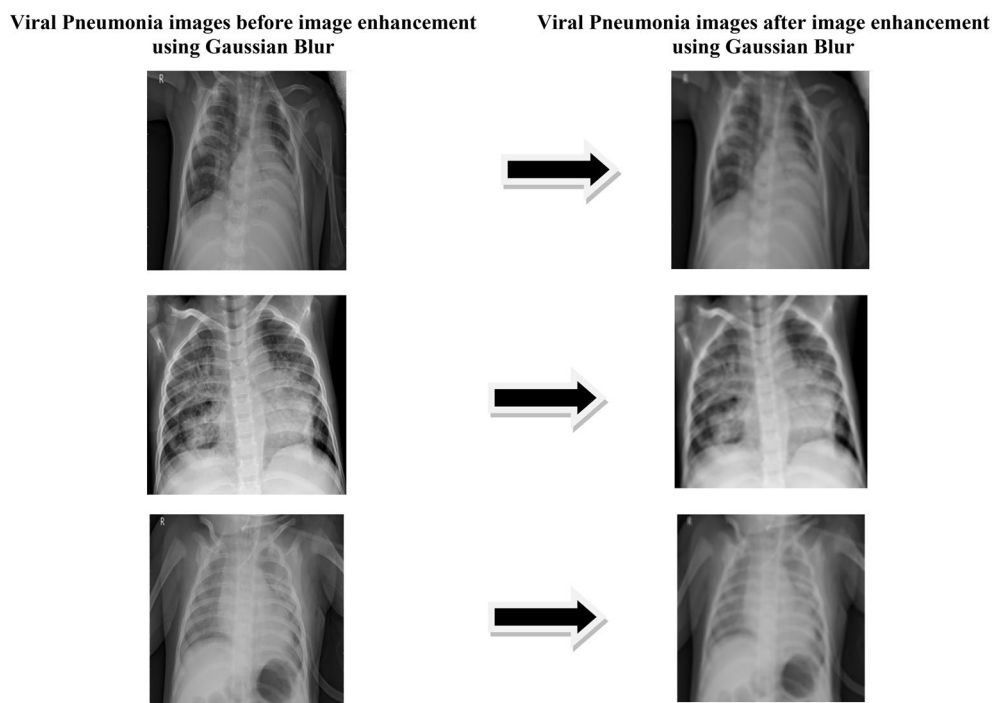


Figure 6. Image enhancement result on viral pneumonia dataset.

Methodology of Gaussian Blur

1. The image pixel take the average value of its surrounding pixels. The center value is considered as average value.
2. Gaussian function is applied to get two dimensional normal distribution as

$$G_{2D}(x,y,\sigma^2) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

3. The value of this function will create convolution matrix/kernel.
4. Gaussian blur is calculated based on the weight matrix.
5. The function is applied on the entire image in the calculation for each image.
6. In the end the Gaussian continuous value is converted into discrete values needed for a kernel.

Algorithm of Gaussian Blur

The CLAHE⁵² and histogram equalization⁵³ techniques are experimented before the selection of Gaussian Blur. The CLAHE define as contrast limiting adaptive histogram equalization which refine the images with high intensity. To improve contrast of image, histogram equalization is used. The other two techniques give less accuracy as compared to the blur technique⁵¹.

Data augmentation

In order to make training data more generalizable and applicable, data augmentation involves transforming images in various ways, such as rotating, flipping, and resizing⁵⁴. Figure 7 shows the images without data augmentation and with data augmentation¹⁹. It increases the size of samples used as training set by applying different techniques written in the Table 2 which help the model to extract features and understand the image. The technique provide good results for enhancing the performance and is used to reduce over fitting⁴⁵. The data augmentation methods is represent in Table 3 with different parameters.

The augmentation algorithms include kernel filters, geometric transformations, random erasing, mixing images, color space augmentations, etc. The above results show that preprocessing of image with data augmentation can increase the precision of classification and reduces the overfitting problem.

Evaluation parameters

Based on the confusion matrix, we will determine the class-wise performance of our model based on the following performance metrics⁵⁵.

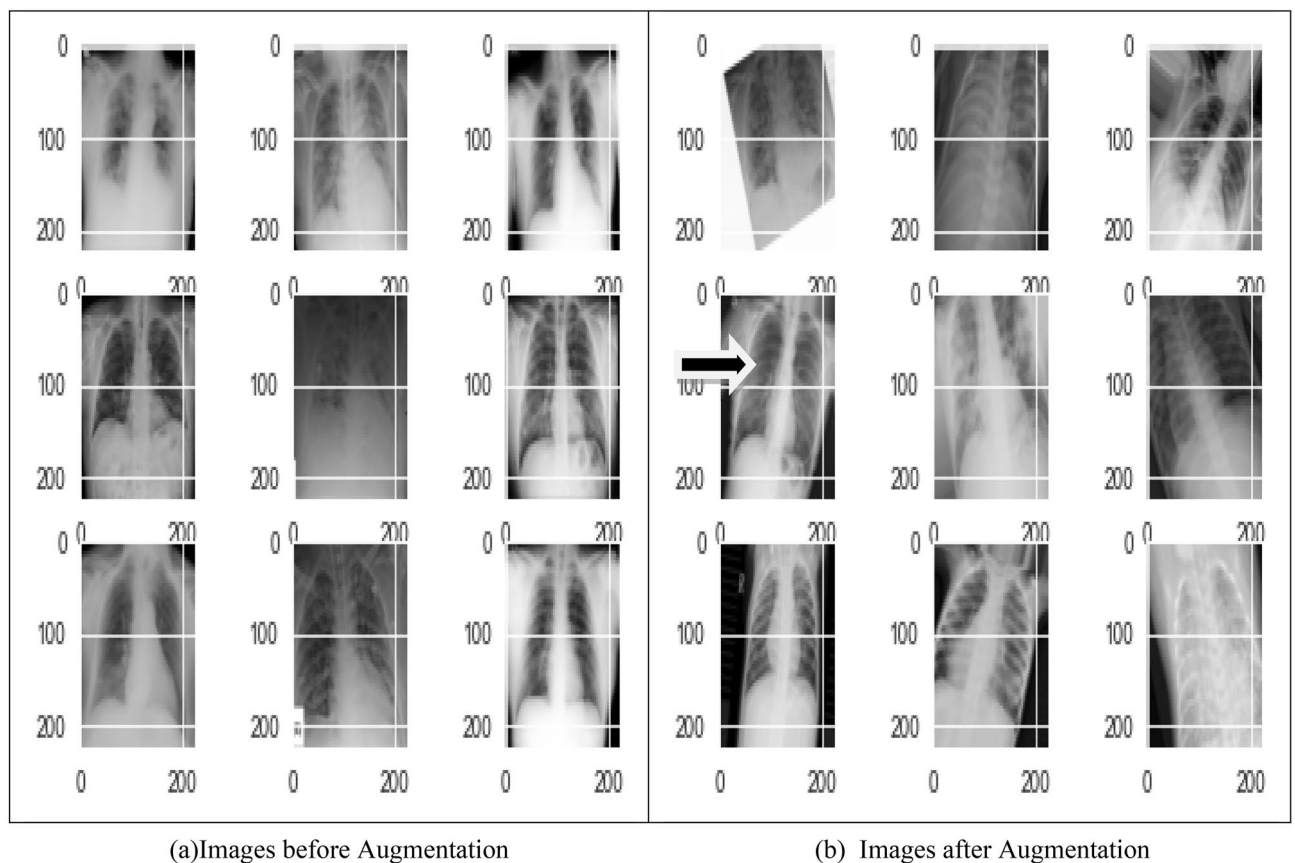


Figure 7. (a) Images before augmentation (b) images after augmentation.

Augmentation method	Value
Rescale	1/255
Rotation range	30
Zoom range	0.2
Width shift range	0.1
Height shift range	0.1
Horizontal flip	True

Table 3. Data augmentation methods.

1. True positive (TP): These are instances in which both the predictive and actual class are true(P).
2. True negative (TN): True negatives arise when both the expected and actual classes are false(N).
3. False negative (FN): These are instances in which the data item's real class is true (P), but the classification model wrongly labels it as false (N).
4. False positive (FP): These are instances when the data item's real class is false (N), but the classification model mistakenly labels it as true (P).
5. Accuracy: Accuracy is a percentage of correct predictions to total predictions. Equation (14) defines the accuracy formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

6. Loss: The difference between the predicted and actual value is the loss. It is a way for calculating loss. Equation (15) depicts the loss formula, where y represents the predicted outcome and \hat{y} represents the model's output

$$L = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \quad (15)$$

7. Execution time: The time the model takes from start to finish of execution.
8. Recall: The percentage of total relevant results properly categorized by the model is referred to as recall.

$$Recall = \frac{TP}{TP + FN} \quad (16)$$

9. Precision: It is the percentage of relevant results in your results. The formula is as follows:

$$Precision = \frac{TP}{TP + FP} \quad (17)$$

10. F-Measure: The F-measure represents the harmonic mean of accuracy and recall. It is determined as follows:

$$F-Measure = \frac{2 * Recall * Precision}{Recall + Precision} \quad (18)$$

Experimental results and discussion

This part represents and analyze the results obtained after performing experiments in three different scenarios. All the proposed approaches have been executed with python using Tensor flow and Keras libraries³⁸. For the analysis of results, dataset is categorized into different ratios as represented in Table 4.

Experimental results without image enhancement

Scenario 1

The following results are based on scenario 1, i.e., dataset ratio of 70:30 in which 70% of images belong to the training set and 30% belong to the testing dataset. After hyperparameter tuning step, the suggested approach is compared to other models. Figure 8 represents the confusion matrix.

Figure 8 represents the confusion-matrix of bifurcation of the dataset. Figure 9 represents the percentage of success for classifying COVID disease in 500 epochs. The proposed model gives 96% accuracy which is better

	Scenario-1	Scenario-2	Scenario-3
Training	70	60	80
Testing	30	40	20

Table 4. Dataset distribution.

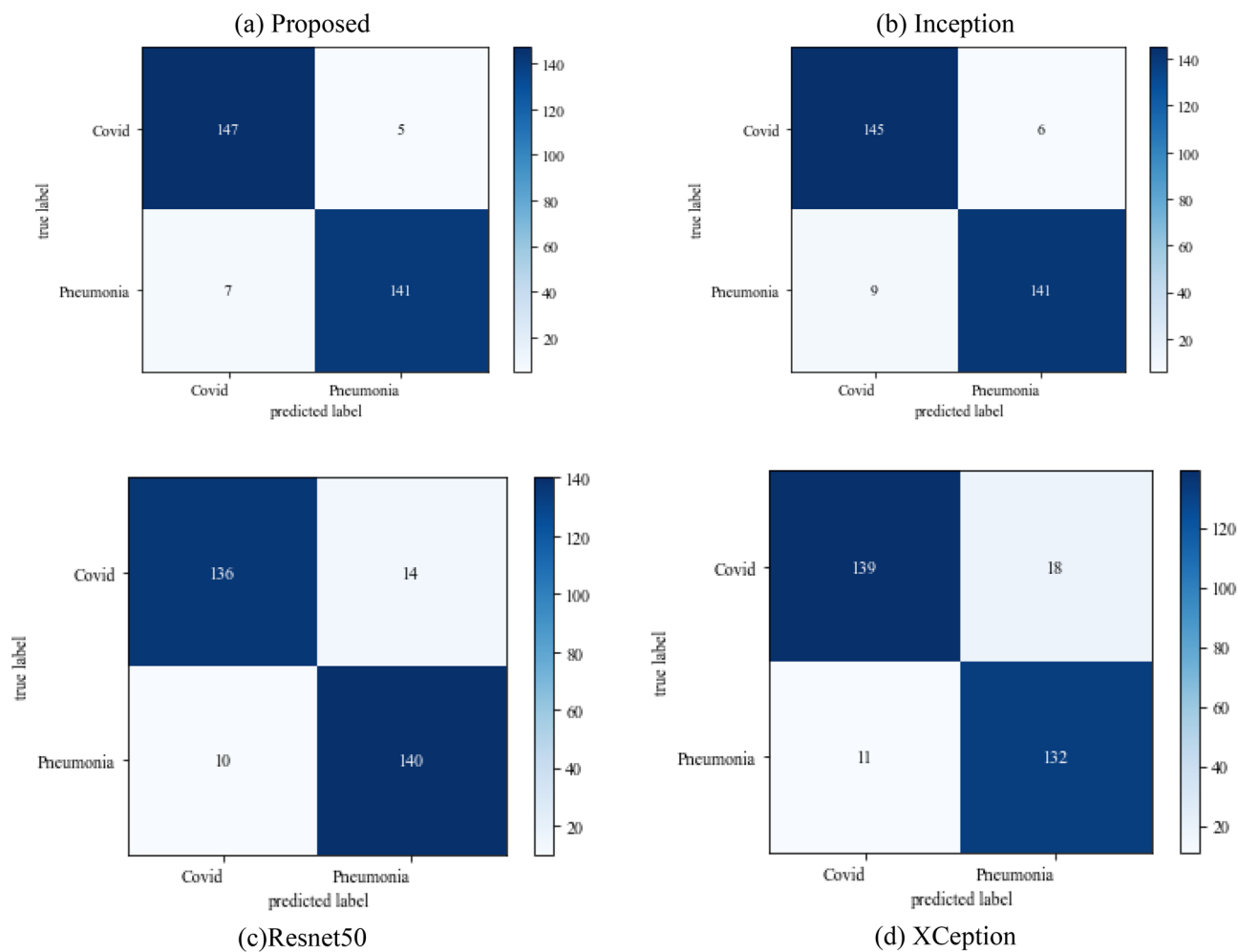


Figure 8. Confusion matrix on scenario 1.

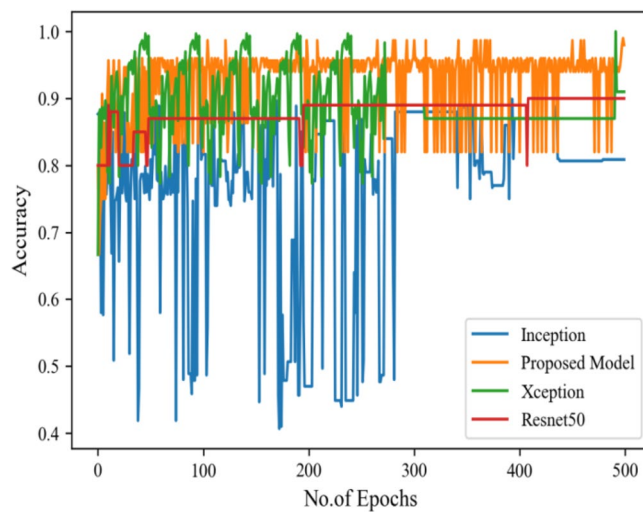


Figure 9. Testing accuracy with 500 epochs.

than other models. The bar graph representation in Fig. 10 sum up the time of execution. The proposed approach takes less time as contrast to other models because it is lightweight and used fewer parameter which makes it faster than other models.

Moving ahead, Table 5 shows the performance metrics obtained in scenario 1, containing the value of F1-score, recall and precision. In this table, the proposed model shows the highest value of F1-score, recall and precision for class COVID and viral-pneumonia. Figure 10 represents the testing loss of each model with 500 epochs. The loss value shows how much error rate is there in the model performance, the resulting graph shows that Resnet50 has high value of loss rate as compared to other models. Figure 11 illustrates the testing loss with 500 epochs for all models.

Scenario 2

The following results are based on scenario 2 with a dataset ratio of 60:40 in which 60% of images belong to the training dataset and 40% belong to testing dataset. Figure 12 shows a confusion matrix that illustrates that the proposed model and Inception model have high true positive value i.e. 141 and 142 as compared to other models. In the confusion matrix true positive value of proposed model is 141 which means the COVID images are correctly classified as COVID and 136 as true negative value which shows the viral pneumonia images correctly classified.

Table 6 displays the f1-score, recall and precision value of models. The model outperforms the Inception model in precision and surpasses the Xception model in recall. The projected model gives the maximum value of precision and recall which makes it better than other models.

Figure 13 represents the result of truly and correctly classified images of viral-pneumonia and COVID. The testing accuracy with Resnet50, Inception and Xception is less as compared to the proposed model. Figure 14 displays the bar graph of the time executed by each model during the execution of the 500 epochs.

The proposed model was executed in 7460 s in total which is less than other transfer learning CNN models. The proposed model was executed in less time because the model architecture have lesser parameters as compared to other models. Figure 15 shows the result of validation loss rate with Resnet50, Inception and Xception. Loss defines how many the wrong predictions were made by the model. The proposed model gives less value of loss rate as compared to other models.

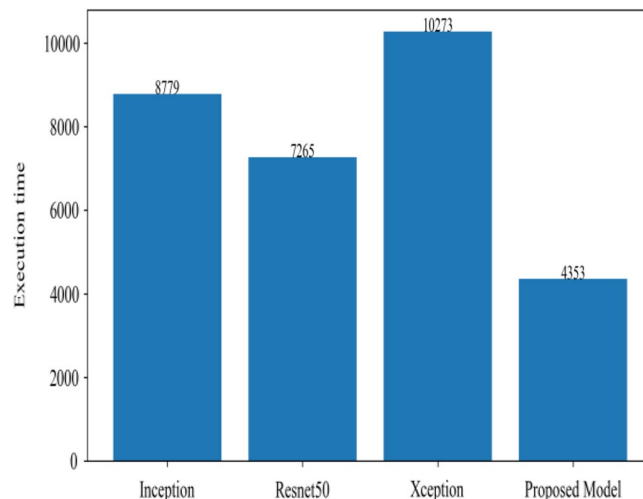


Figure 10. Execution time of each model.

Model	Labels	Precision	Recall	F1-score
InceptionV3	COVID-19	0.96	0.89	0.92
	Viral_pneumonia	0.90	0.96	0.93
Resnet50	COVID-19	0.90	0.89	0.89
	Viral_pneumonia	0.85	0.88	0.86
Xception	COVID-19	0.90	0.86	0.88
	Viral_pneumonia	0.92	0.91	0.91
Proposed model	COVID-19	0.97	0.95	0.96
	Viral_pneumonia	0.95	0.97	0.96

Table 5. Model results with scenario 1.

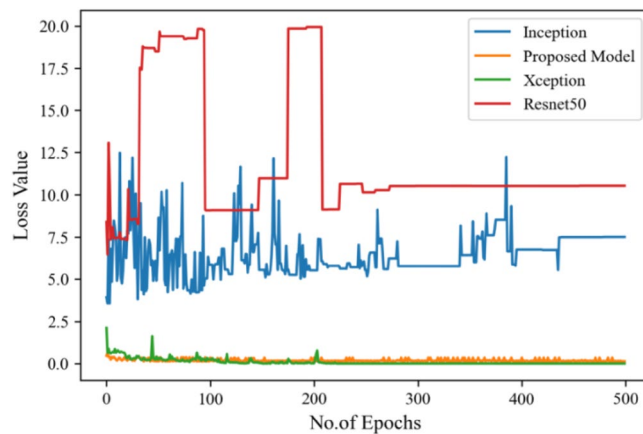


Figure 11. Testing loss with 500 epochs.

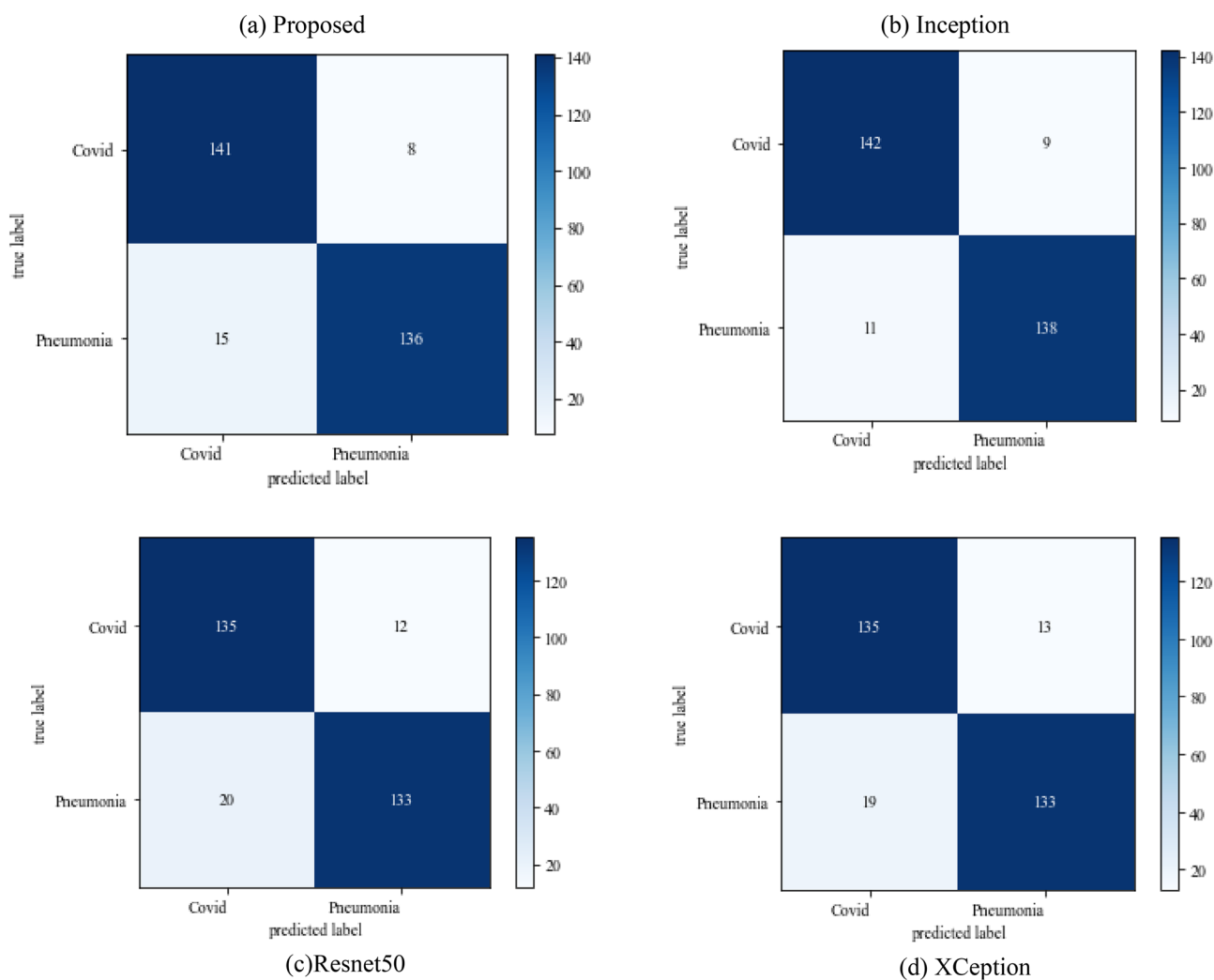


Figure 12. Confusion_matrix on scenario-2.

Scenario 3

The following results are based on scenario 3 on the dataset ratio 80:20 in which 80% of images belong to the training dataset and 20% belong to testing dataset. The confusion matrix which shows the result of model performance in predicting true images of diseases has been illustrated in Fig. 16

Model	Labels	Precision	Recall	F1-score
InceptionV3	COVID-19	0.93	0.92	0.92
	Viral_pneumonia	0.92	0.94	0.93
Resnet50	COVID-19	0.84	0.87	0.85
	Viral_pneumonia	0.89	0.86	0.87
Xception	COVID-19	0.94	0.91	0.92
	Viral_pneumonia	0.92	0.94	0.93
Proposed model	COVID-19	0.96	0.95	0.95
	Viral_pneumonia	0.95	0.96	0.95

Table 6. Model results on scenario 2.

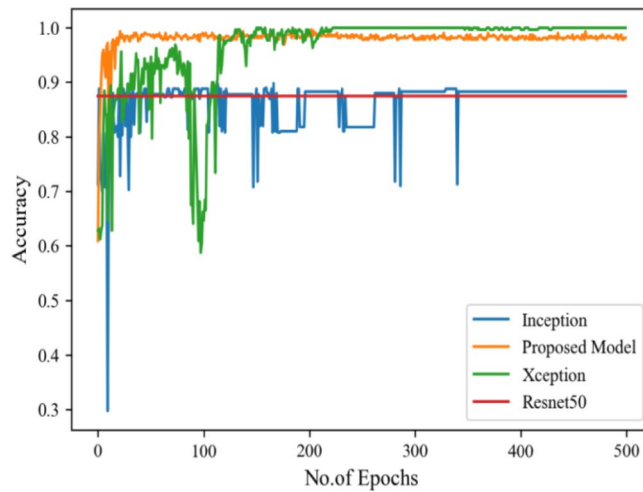


Figure 13. Testing accuracy with 500 epochs.

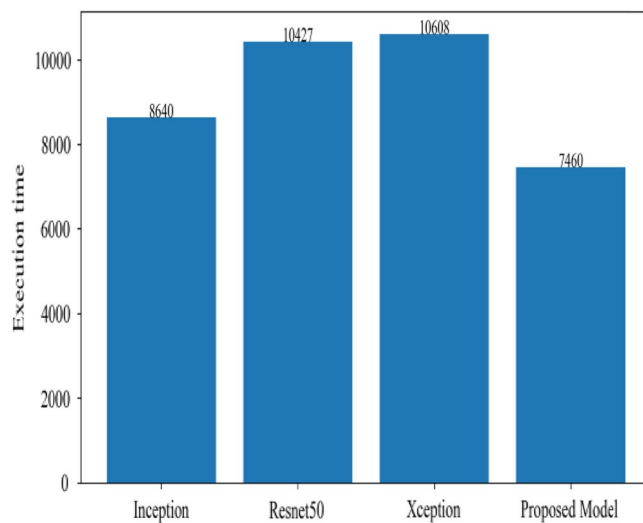


Figure 14. Execution time by each model.

Figure 16 depicts the findings of the analysis confusion matrix, as well as the values related to performance metrics of the transfer learning model as well as suggested model. Table 7 displays the significance of the findings in terms of accuracy, recall, and f1-score of Inceptionv3, Resnet50, Xception, and the proposed model while using Scenario 3. Figure 17 represent the increasing success rate of classified data of proposed model as compared to

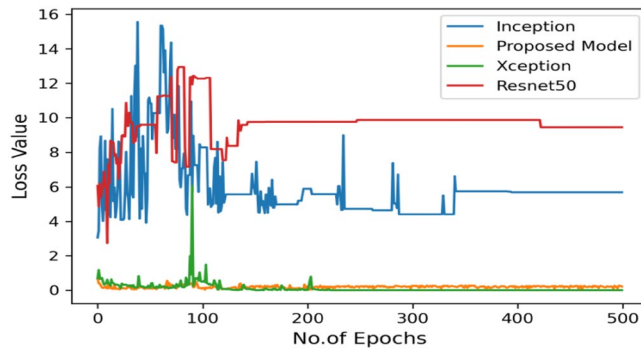


Figure 15. Testing loss with 500 epochs.

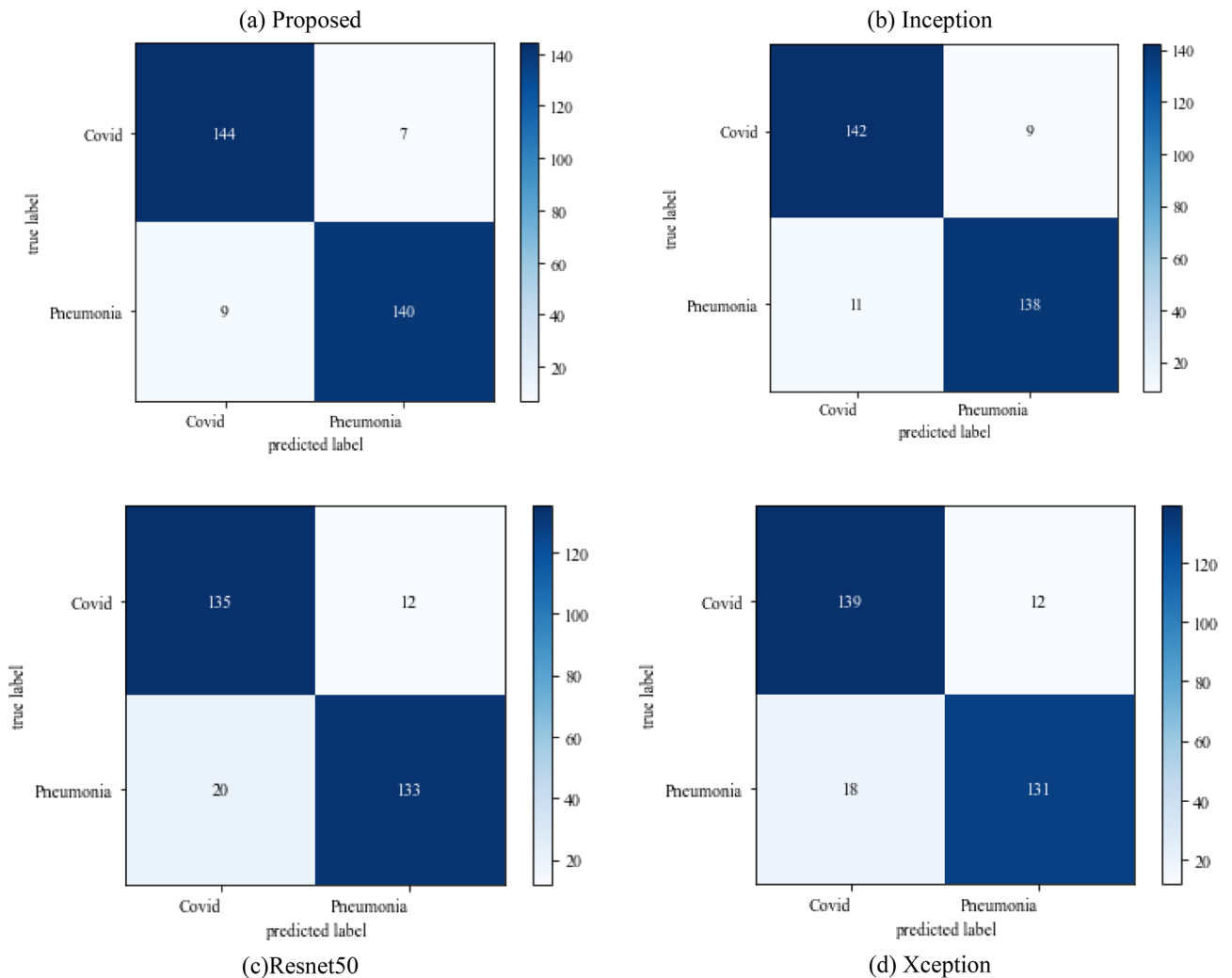


Figure 16. Confusion matrix on scenario 3.

three models with every epoch. The testing accuracy of Resnet50 is 0.89, Inception is 0.93, Xception is 0.74 and the proposed model is 0.96.

Figure 18 shows the bar graph of the time which define the completion of task by each model in 500 epochs. This bar graph shows the results of each model with 80:20 ratio dataset. The proposed model executed in 7114 s in total. Figure 19 shows the result of wrongly classified images rate by each model and the graph represent the different peak of loss rate with each epoch. The Resnet50 have highest value of loss as compared to other models.

Model	Labels	Precision	Recall	F1-score
InceptionV3	COVID-19	0.92	0.91	0.91
	Viral_pneumonia	0.90	0.96	0.93
Resnet50	COVID-19	0.90	0.86	0.88
	Viral_pneumonia	0.85	0.88	0.86
Xception	COVID-19	0.91	0.89	0.90
	Viral_pneumonia	0.92	0.94	0.93
Proposed model	COVID-19	0.96	0.97	0.96
	Viral_pneumonia	0.95	0.96	0.95

Table 7. Model results on scenario 3.

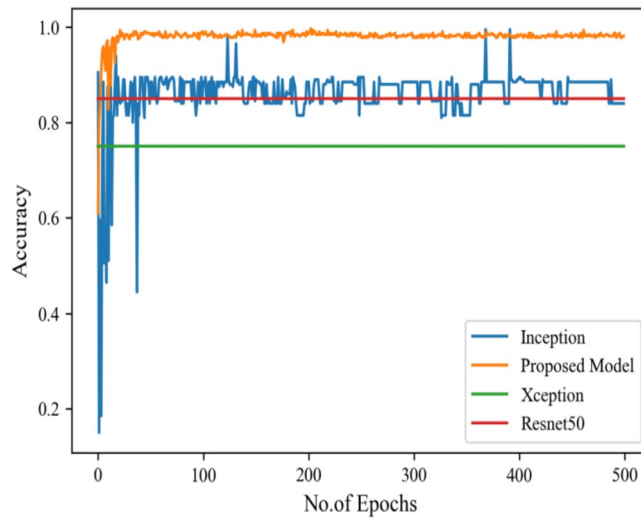


Figure 17. Testing accuracy with 500 epochs.

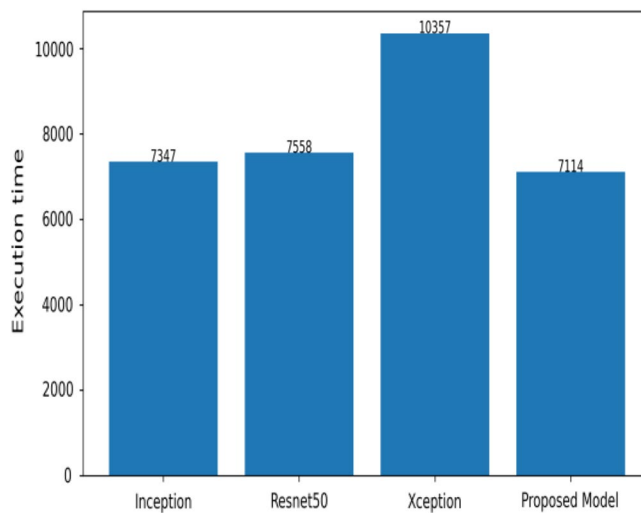


Figure 18. Execution time by each model.

Experimental results with image enhancement

The following results are taken with Gaussian Blur images for 500 epochs, and maximum accuracy of 98% was observed. With every epoch, the accuracy rate of our proposed model with image enhancement gets improved. The functioning of CNN model with image enhancement images got improved in every epoch, the model uses smoothed and less noise image which increase its accuracy from 96 to 98%.

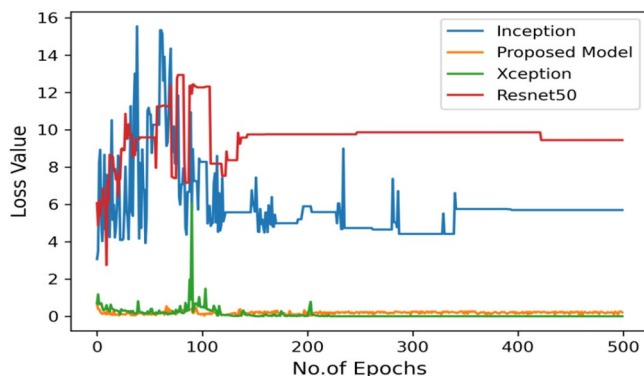


Figure 19. Validation loss with 500 epochs.

Scenario 1

The following results are based on scenario 1, i.e., dataset ratio of 70:30 in which 70% of images belong to the training set and 30% belong to the testing part of image enhancement dataset.

Figure 20 shows the confusion matrix of image enhancement dataset in scenario 1. The confusion matrix shows the truly identified images by model and help in deep analysis of model performance. Figure 21 represents the true positive and true negative correctly identified classes of dataset in 500 epochs. The proposed model gives 98% accuracy which is better than other model's accuracy. The bar graph shows in Fig. 22 show the execution time of the model. The proposed model architecture contains less parameter which decrease the execution time.

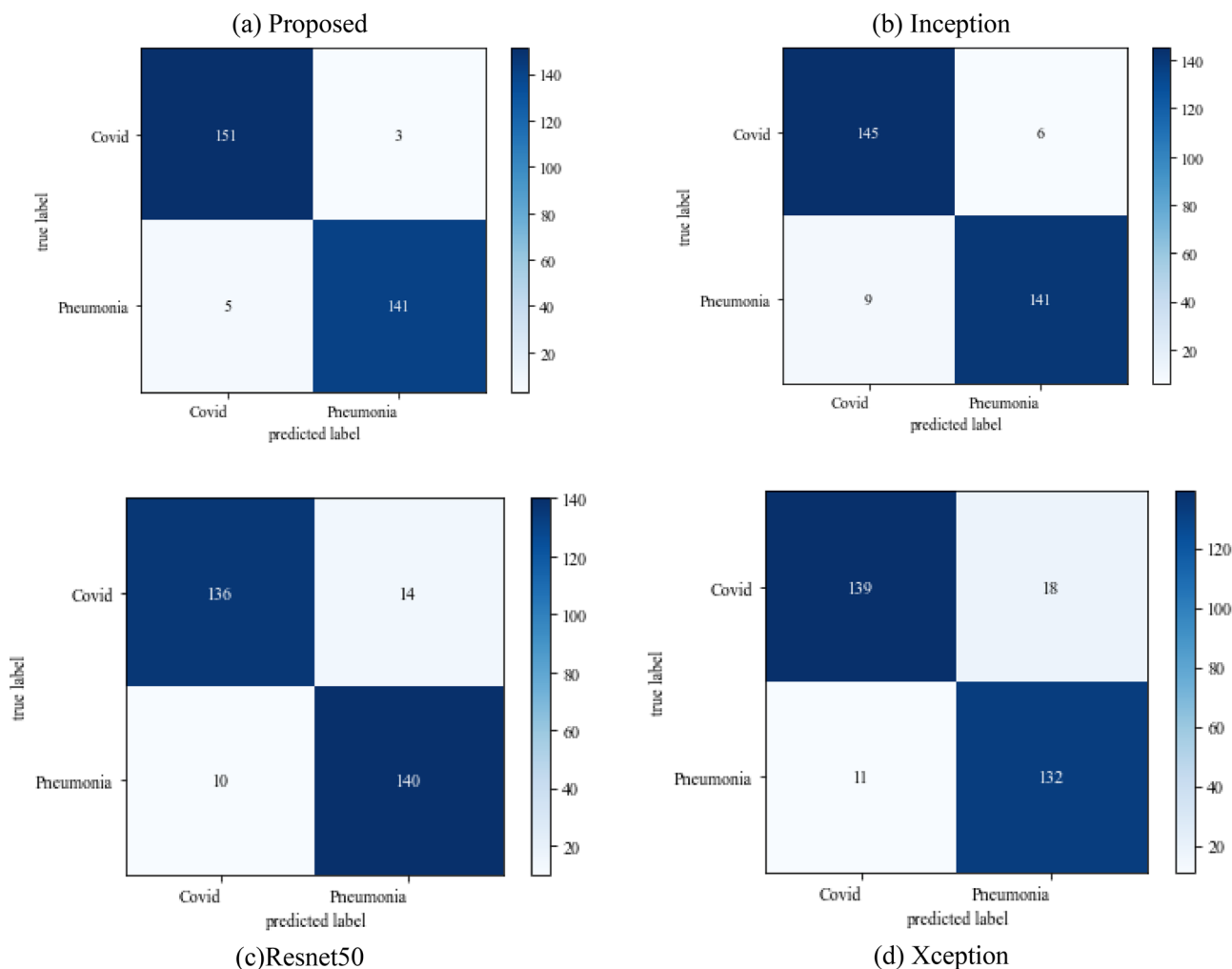


Figure 20. Confusion matrix on scenario 1.

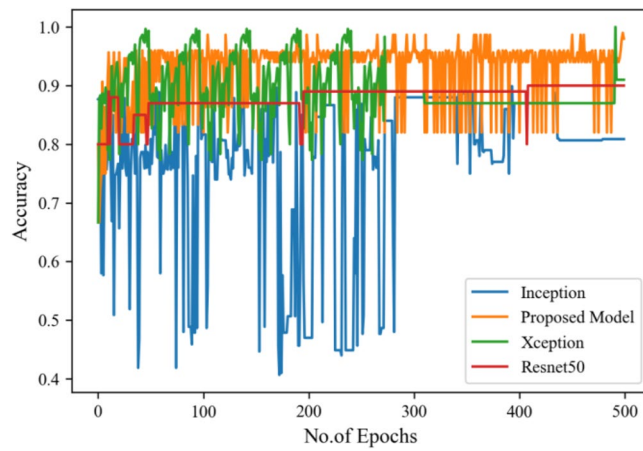


Figure 21. Testing accuracy with 500 epochs.

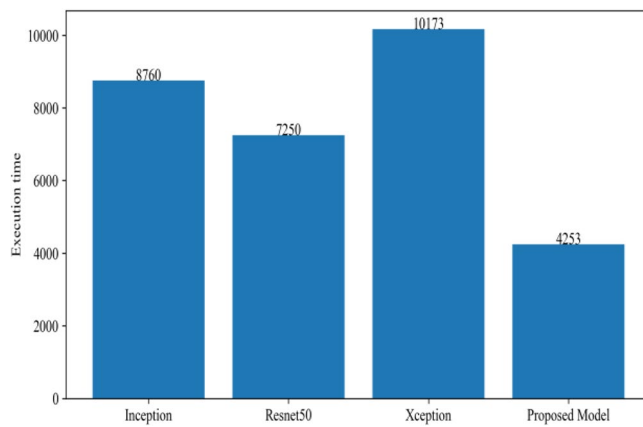


Figure 22. Execution time by each model.

Moving ahead, Table 8 shows performance metrics obtained in scenario 1. Accuracy can be misleading if used with imbalanced dataset and therefore the other metrics such as f1-score, recall and precision useful are for evaluation. Table 6 is the summarization of the results of each model. Figure 23 represents the loss rate of each model with 500 epochs. The Loss value shows how much error rate is there in the model's performance, the resulting graph shows that Resnet50 has high value of loss rate as compared to other models.

Scenario 2

The following results are based on scenario 2 with a dataset ratio of 60:40 in which 60% of images belong to the training dataset and 40% belong to testing dataset. The classifier matrix has been portrayed in Fig. 24, which shows the true positive value of proposed model is 151 which is high than others. Table 9 give the performance

Model	Labels	Precision	Recall	F1-score
InceptionV3	COVID-19	0.96	0.89	0.92
	Viral_pneumonia	0.90	0.96	0.93
Resnet50	COVID-19	0.93	0.92	0.92
	Viral_pneumonia	0.92	0.94	0.93
Xception	COVID-19	0.90	0.89	0.89
	Viral_pneumonia	0.85	0.88	0.86
Proposed model	COVID-19	0.97	0.97	0.97
	Viral_pneumonia	0.97	0.98	0.98

Table 8. Model results with scenario 1.

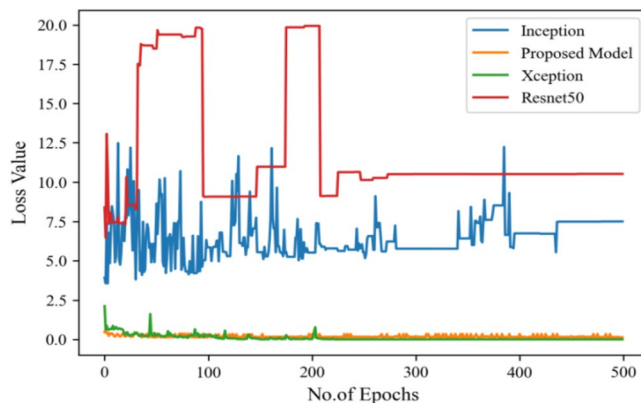


Figure 23. Testing loss with 500 epochs.

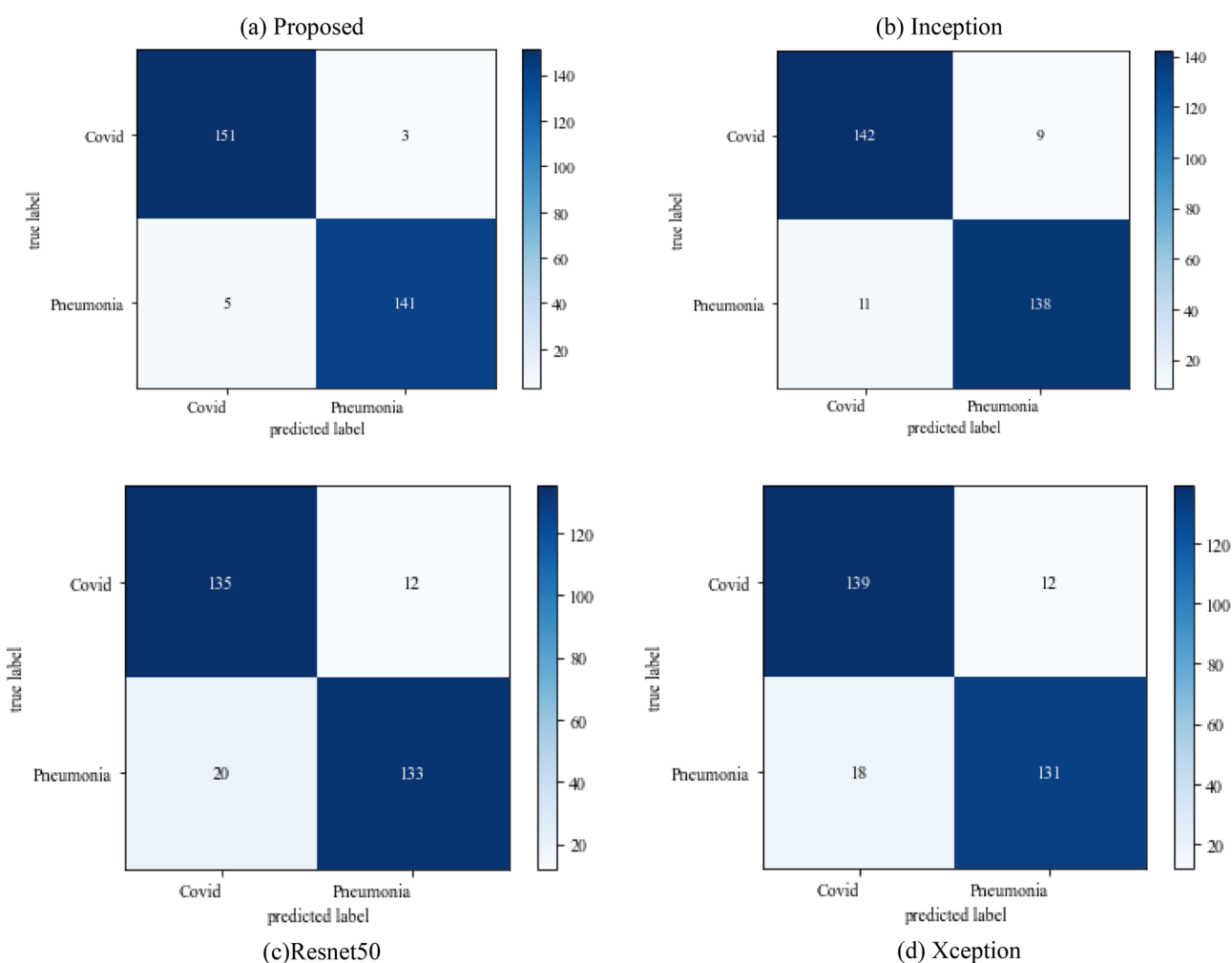


Figure 24. Confusion matrix on scenario 2.

of models which shows the proposed approach model gives the better value of precision and recall which makes it better than other models.

Figure 25 illustrated the percentage of success in classifying COVID and viral-pneumonia class correctly. The testing accuracy with Resnet50, Inception and Xception is less as compared to the this paper model. It gives high classification result of 98% as compared to other models with image enhancement images. Figure 26 shows the bar graph of the time executed by each model during the execution of the 500 epochs. The proposed model

Model	Labels	Precision	Recall	F1-score
InceptionV3	COVID-19	0.91	0.89	0.90
	Viral_pneumonia	0.92	0.94	0.93
Resnet50	COVID-19	0.90	0.89	0.89
	Viral_pneumonia	0.85	0.88	0.86
Xception	COVID-19	0.90	0.86	0.88
	Viral_pneumonia	0.92	0.91	0.91
Proposed model	COVID-19	0.96	0.97	0.96
	Viral_pneumonia	0.97	0.98	0.97

Table 9. Model results in scenario 2.

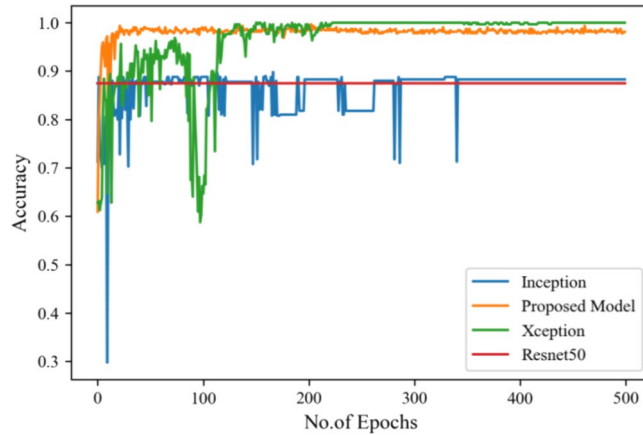


Figure 25. Testing accuracy with 500 epochs.

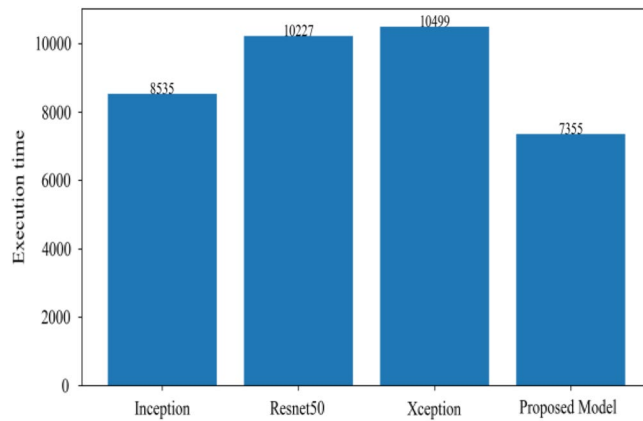


Figure 26. Execution time by each model.

was executed in 7335 s in total which is less than other CNN models. Figure 27 shows the result of validation loss and also the error rate. When pitted against other models, the proposed model exhibits a reduced loss rate.

Scenario 3

The following results are based on scenario 3 on the dataset ratio 80:20 in which 80% of images are belonging to the training dataset and 20% belong to testing dataset. Figure 28 shows the confusion matrix which has obtained values of performance metrics of transfer learning model and proposed model.

Table 10 shows the precision, recall and f1-score values of Inceptionv3, Xception Resnet50 and the proposed model on the dataset in scenario 3 with 80% of the photos are for training, while 20% are for testing. Figure 29 represents that the testing accuracy of Resnet50 is 0.89, Inception is 0.93, Xception is 0.74 and the proposed

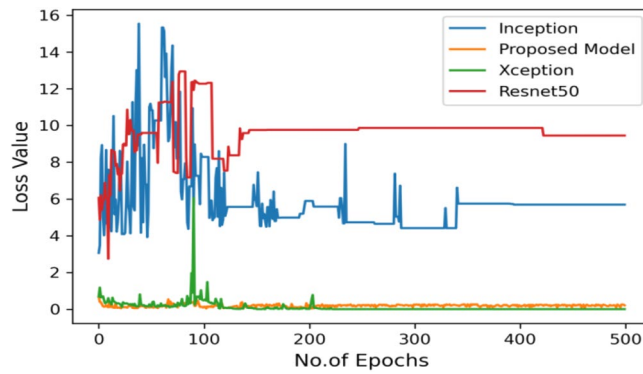


Figure 27. Testing loss with 500 epochs.

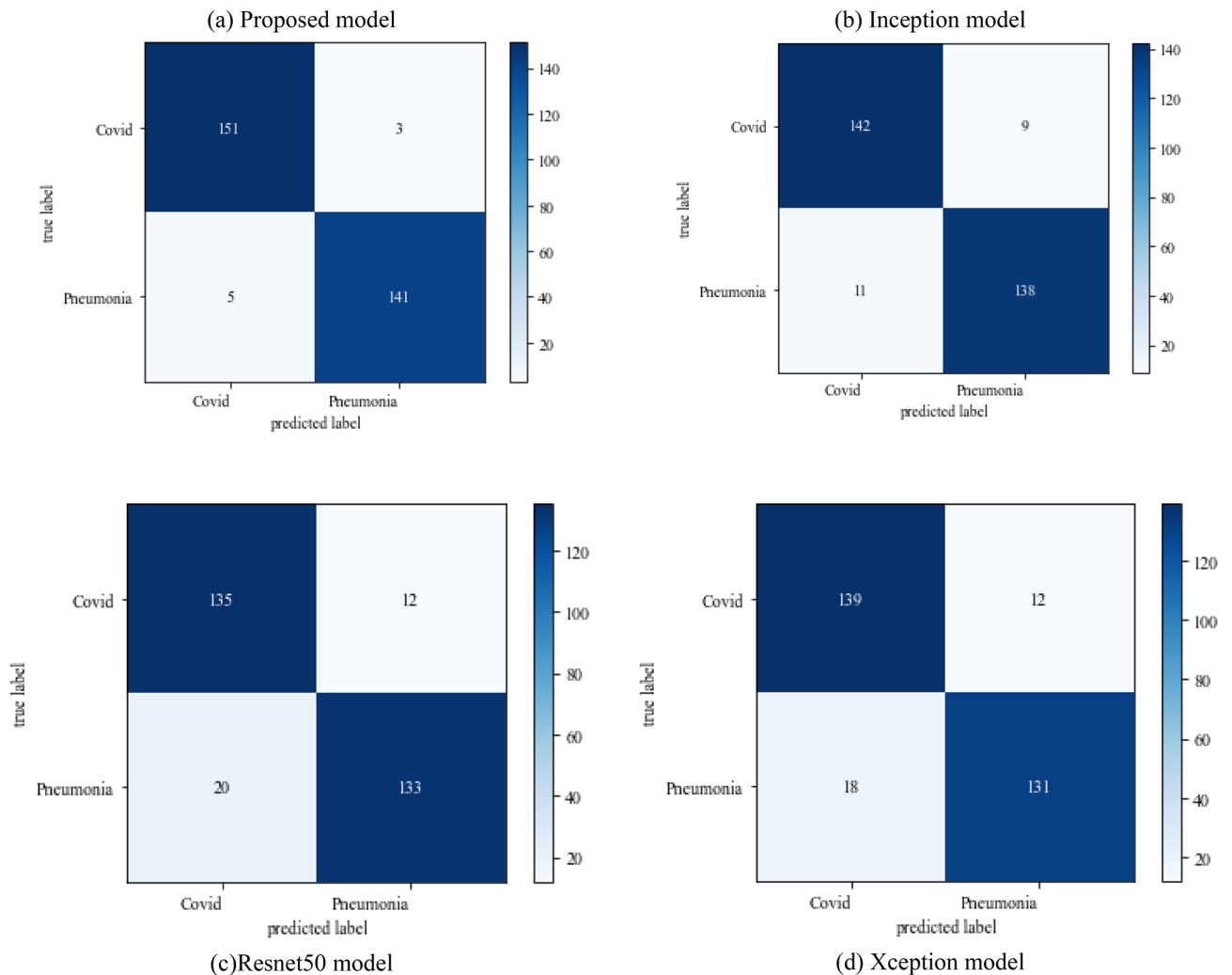


Figure 28. Confusion matrix on scenario 3.

approach is 0.98. When compared to other models, the Xception produces lower value results. Figure 30 shows the execution time of each model in 500 epochs.

Figure 31 exhibits the validation loss outcome along with the result analysis, providing the count of actual and predicted labels for all classes.

Precision values show that out of all the non-COVID 19 infected cases, how many were identified correctly by the model. Figure 32 shows the overall precision value of Resnet50, Inception, Xception and the proposed approach. Figure 32a represents the values for class COVID 19 in Resnet50 is 88, Inception gives a value of 93.6,

Model	Labels	Precision	Recall	F1-score
InceptionV3	COVID-19	0.92	0.91	0.91
	Viral_pneumonia	0.90	0.96	0.93
Resnet50	COVID-19	0.90	0.86	0.88
	Viral_pneumonia	0.85	0.88	0.86
Xception	COVID-19	0.91	0.89	0.90
	Viral_pneumonia	0.92	0.94	0.93
Proposed model	COVID-19	0.98	0.97	0.97
	Viral_pneumonia	0.98	0.98	0.98

Table 10. Model results on scenario 3.

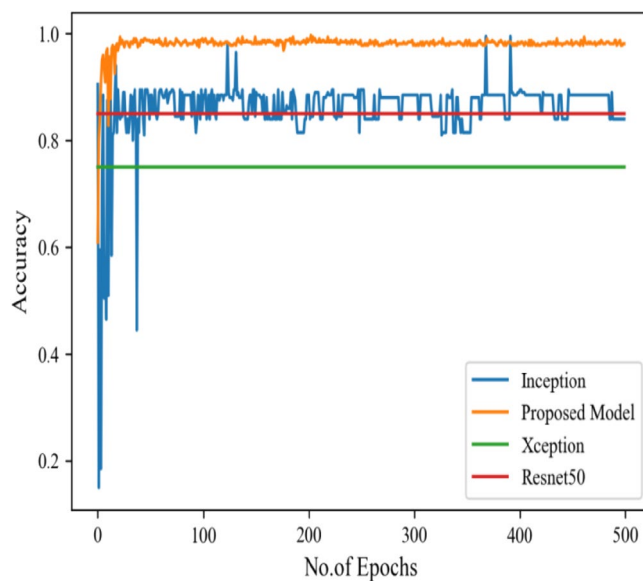


Figure 29. Testing accuracy with 500 epochs.

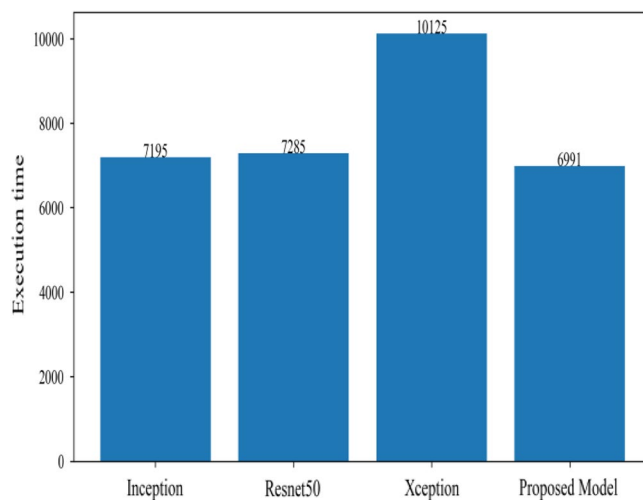


Figure 30. Execution time by each model.

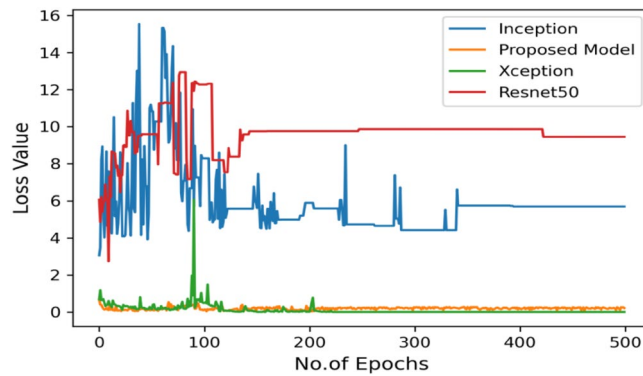
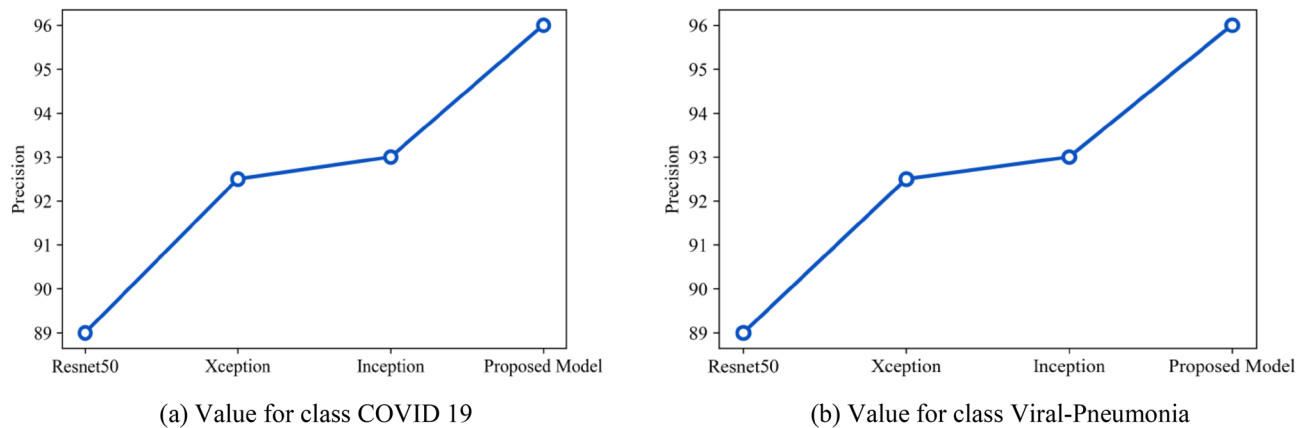


Figure 31. Validation loss with 500 epochs.



(a) Value for class COVID 19

(b) Value for class Viral-Pneumonia

Figure 32. (a) Value for class COVID 19, (b) value for class viral-pneumonia.

Xception gives a value of 91.6 and proposed model is 96.3. Figure 32b shows the values for class Viral pneumonia as Resnet50 is 86.5, Inception gives a value of 90.6, Xception gives a value of 92 and the proposed model is 97.

Aggregation of three scenarios

This section presents the average of all the results obtained in the above mentioned three scenarios. The results are the average of the model which includes the three phases: image enhancement, data augmentation and hyperparameter tuning. The average value of recall, precision, f1-measure and accuracy based on the average of all results with image enhancement dataset.

Recall value shows out of all the COVID 19 patients, how many were identified correctly by the model. Figure 33 shows the overall recall value (in percentage) of Resnet50, Inception, Xception and the proposed approach. Figure 33a represents the values for class COVID 19 as Resnet50 is 95.6, Inception gives a value of 90, Xception gives a value of 88.6 and proposed model is 95.3. Figure 33b shows the values for class Viral pneumonia as Resnet50 is 87, Inception gives a value of 95, Xception gives a value of 93 and proposed model is 98.

Unfortunately, it is not feasible to maximize both accuracy and recall simultaneously. So, there is another metric available for convenience called F-measure. Figure 34a shows f1-score values for class COVID 19 as Resnet50 is 87.3%, Inception gives a value of 91%, Xception gives a value of 90% and proposed model is 96%. Figure 34b shows the values for class Viral pneumonia as Resnet50 is 86.3%, Inception gives a value of 93%, Xception gives a value of 92% and for the proposed model it is 97%.

Figure 35 shows the average accuracy of all models based on the different experiments. The average accuracy of Resnet50 is 0.89, Inception accuracy is 0.93, the accuracy of Xception is 0.92 and proposed model accuracy is 0.98. The proposed model perform better in comparison to other models as shown in Figure 36 as ROC curve of models.

Table 11 represent the evaluation metrics such as sensitivity, specificity, positive and negative likelihood to enhance the comprehensibility of the model's performance. The proposed model give better results than other transfer learning models.

It can be clearly gauged from the above result that the model proposed here is better than various other models in terms of various parameters. The outperformance of draft model can be credited to the various modifications to the CNN architecture, which can be further summarized by the following points.

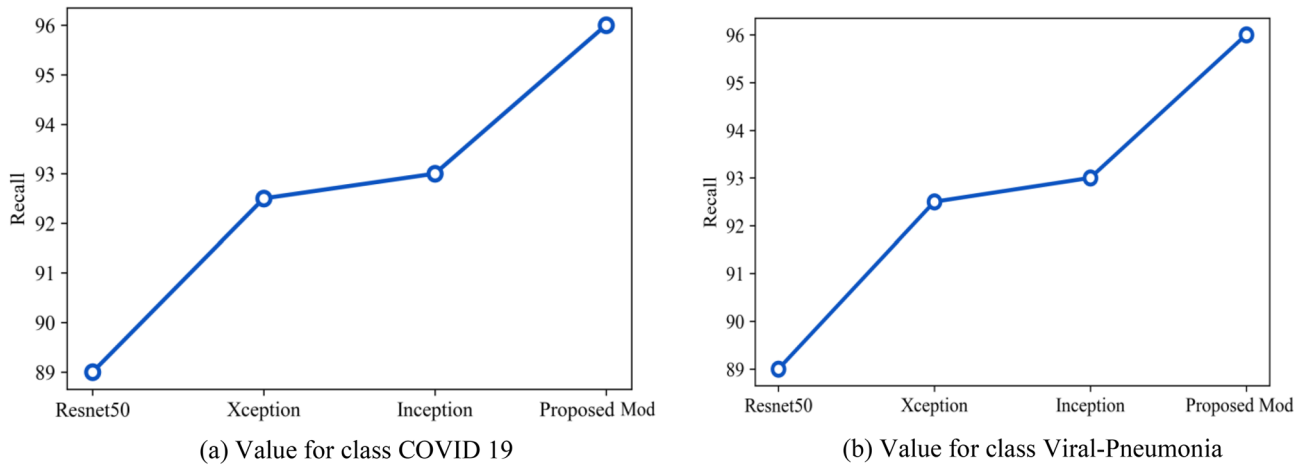


Figure 33. (a) Value for class COVID 19, (b) value for class viral-pneumonia.

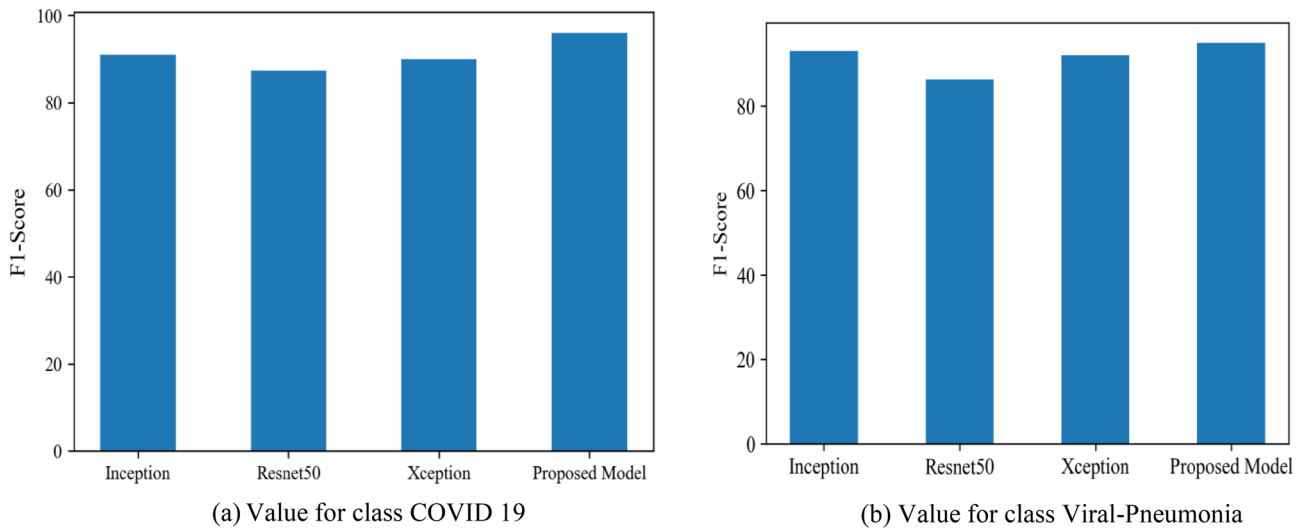


Figure 34. (a) Value for class COVID 19, (b) value for class viral-pneumonia.

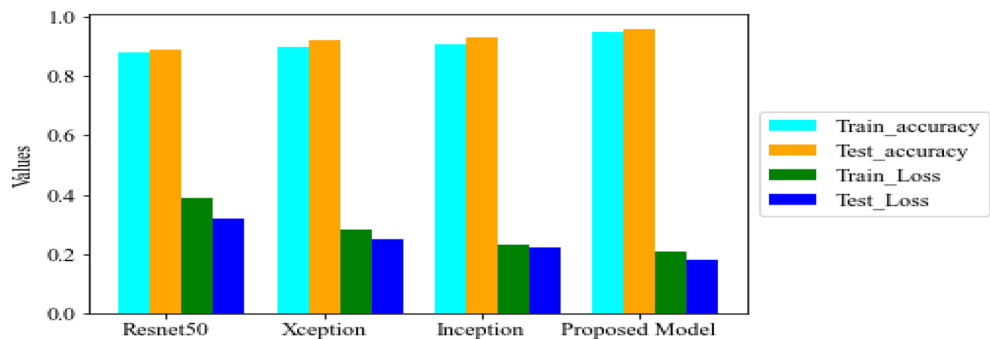


Figure 35. Average accuracy and loss.

1. The proposed model have considered three different image enhancement techniques that are Gaussian Blur, CLAHE, Histogram equalization. The Gaussian Blur removes the noise and intensity of images. Figure 37 show the image enhancement techniques images i.e. CLAHE, Gaussian Blur and Histogram equalization, out of which image enhancement using Gaussian blur technique leads to better classification accuracy as compared to other image enhancement techniques. The classification results obtained by using image enhance-

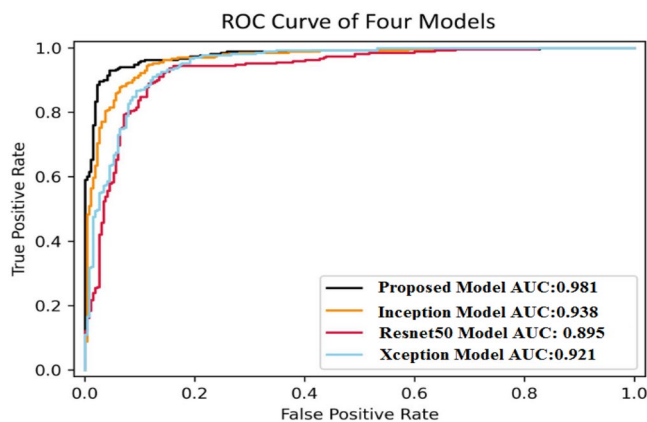


Figure 36. ROC curve of models.

Model	Sensitivity	Specificity	Positive likelihood ratio	Negative likelihood ratio
Xception	0.915	0.918	11.56	0.0789
Resnet50	0.871	0.906	10.54	0.1256
Inception	0.928	0.935	15.16	0.0768
Proposed model	0.972	0.954	21.26	0.0285

Table 11. Comparison table of models.

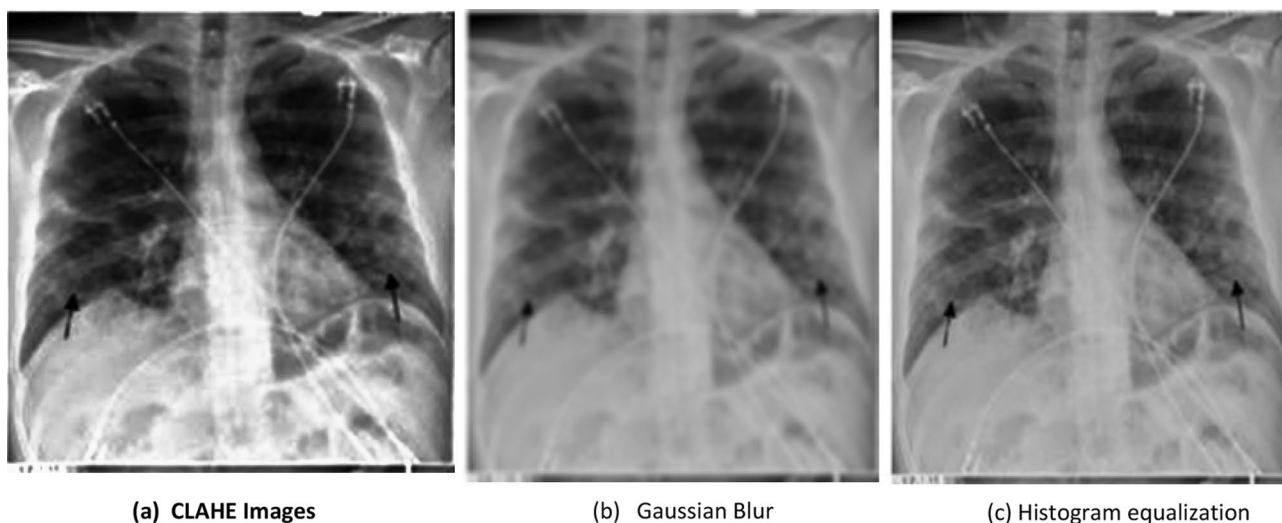


Figure 37. Image enhancement techniques based images.

ment using Gaussian blur is 98%, while image enhancement using CLAHE and histogram equalization give 94% and 93% classification accuracy, respectively.

Table 12, represent the comparative analysis of all three image enhancement techniques with parameters accuracy and loss rate.

2. Authors have also used the concept of data augmentation. It used to large the quantity data by adding slightly modify samples of data images. It can reduce the overfitting of model and also increase classification value by increasing the training data. The below figure clearly depicts the effect of image augmentation. The following two figures show the successful rate and loss rate of draft model before and after data augmentation. The accuracy of the model by using 500 epochs is shown in Fig. 38. And loss of value on 500 epochs is shown in Fig. 39.

Enhancement technique	CLAHE images	Gaussian Blur	Histogram equalization
Accuracy	94%	98%	93%
Loss rate	0.65	0.23	0.74

Table 12. Comparative analysis of image enhancement techniques.

Case No	Loss function	Optimizer	Dropout value	No. of layers (Conv)	Learning rate	Kernel size	Accuracy (%)
1	Sparse categorical cross entropy	RMSprop	0.4	3 layer	0.000001	3 × 3	96.6
2	Binary cross entropy	Adam	0.4	3 layer	0.000001	3 × 3	45
3	Sparse categorical cross entropy	Adam	0.4	3 layer	0.0001	3 × 3	96.3
4	Binary cross entropy	RMSprop	0.4	3 layer	0.000001	3 × 3	50
5	Sparse categorical cross entropy	SGD	0.4	3 layer	0.01	3 × 3	96
6	Binary cross entropy	SGD	0.4	3 layer	0.01	3 × 3	70
7	Binary cross entropy	Adam	0.7	3 layer	0.000001	3 × 3	42
8	Binary cross entropy	RMSprop	0.7	3 layer	0.0001	3 × 3	50
9	Sparse categorical cross entropy	Adam	0.7	3 layer	0.000001	3 × 3	42
10	Binary cross entropy	SGD	0.7	3 layer	0.01	3 × 3	50
11	Sparse categorical cross entropy	RMSprop	0.7	3 layer	0.0001	3 × 3	45
12	Sparse categorical cross entropy	SGD	0.7	3 layer	0.01	3 × 3	50

Table 13. Experiment for hyperparameters tuning.

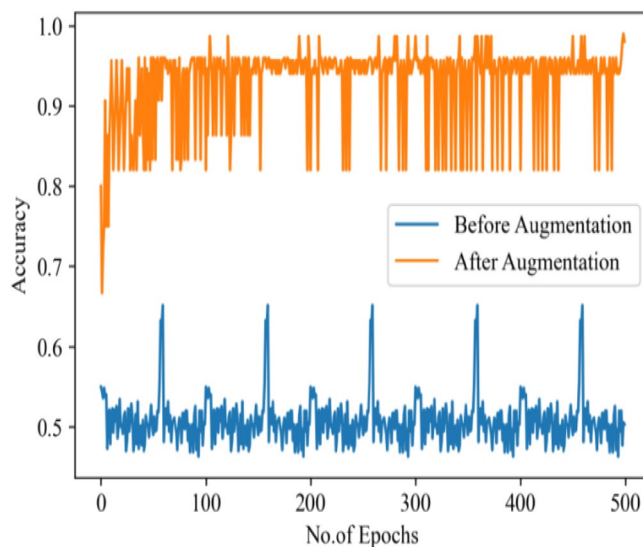


Figure 38. Accuracy value on 500 epoch.

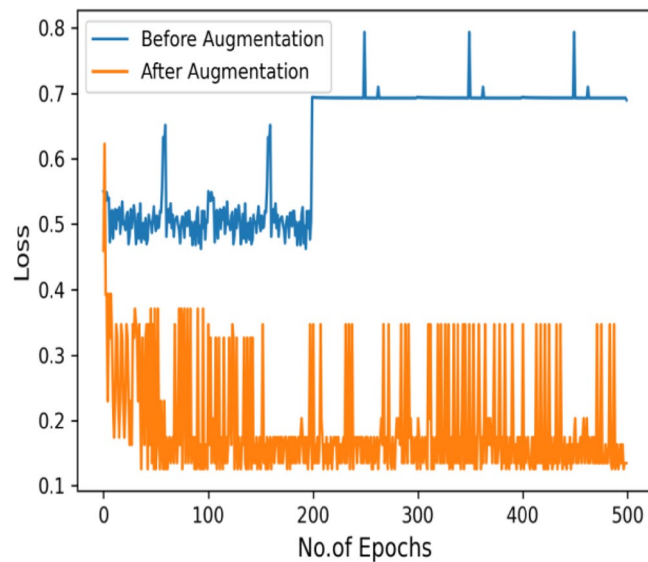


Figure 39. Loss value on 500 epoch.

3. And lastly, authors have tuned the hyperparameters and have made an effort to get the best parameters for increasing the precision and accuracy of classification. The hyperparameters¹⁹ are tuned to find better set of parameters fitted for the proposed model. The tuning of hyperparameters contain the number of neurons, epochs, optimizer, dropout, kernel size, no. of filters, learning rate and activation function. Table 13 shows the experiments conducted on the proposed model using two loss function i.e. Sparse-categorical cross_entropy (LF1) and Binary_cross_entropy (LF2) to adjust all the parameter in right direction. The loss function calculates the difference between the predicted values and the actual target values, providing a feedback signal to update the model's parameters through backpropagation and gradient descent. The choice of loss function depends on the desired behavior with respect to outliers in the data. Minimizing the loss implies that the model's predictions are becoming closer to the true values, leading to improved accuracy.

The confusion matrix of hyper parameter tuning cases is shown in Fig. 40. For the proposed model, The best hyperparameter case 3 selected for better accuracy rate. The results before hyper parameter tuning is 94% which is 2% less than the results after hyperparameter tuning.

Comparative analysis of proposed model with state of the art approaches

Table 14 represents the performance comparison of our proposed model with existing approaches. It can be observed that the proposed model has outperformed all the other approaches.

Conclusion

In current times, deep learning models have come into existence and are playing a massive role in the development of various computer aided disease diagnosis systems. In this research, an attempt has been made to develop a concise CNN model by using lesser number of parameters (no. of layers, kernel size, optimizer, activation function) so as to reduce the execution time while obtaining a better classification accuracy for the diagnosis of COVID-19. The experiments have been performed on various X-ray images, using Gaussian Blur as an image enhancement technique and image augmentation in the image preprocessing stage. The performance of the proposed model is also compared with existing transfer learning models viz. Xception, Resnet50 and inception. The results clearly indicate the outperformance of the proposed model in terms of various parameters (accuracy, loss rate, precision, recall and f1-score). In near future, the proposed model can also be implemented for diagnosis of various other diseases and other different datasets of COVID-19. To reduce the dimensionality of data the various nature inspired meta-heuristic algorithms can be deployed to select the most dominant features while removing the redundant or less important features from the images so as to reduce the computation time and increasing the accuracy of classification.

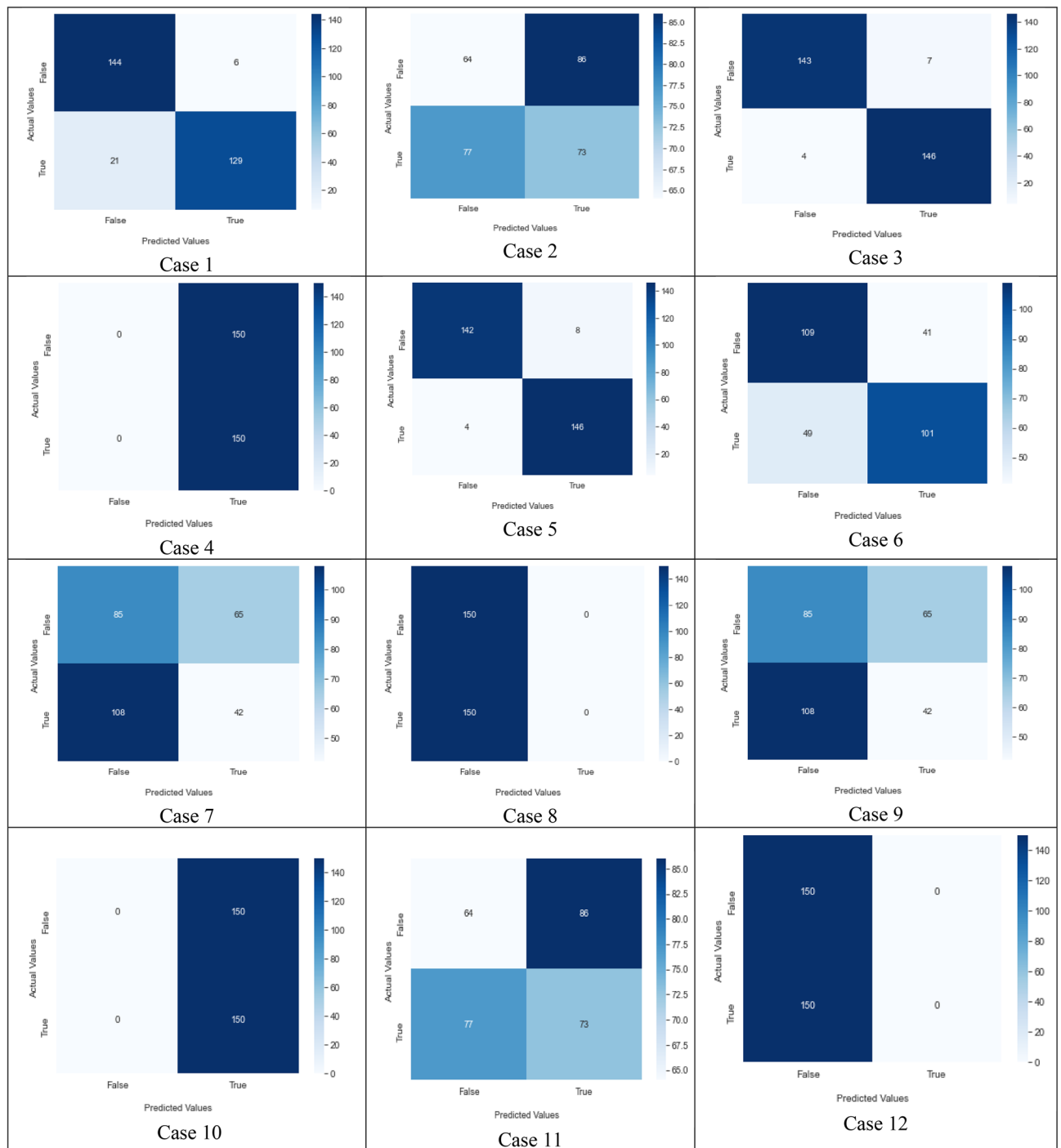


Figure 40. Confusion matrix of all hyperparameter tuning cases.

Author	CNN Dark net ⁵⁶	VGG19 ⁶	Mobile Net v2 ⁵⁷	Inception ⁵⁸	Xception Net ⁵⁹	Mobilenetv2 and VGG16 ²⁰	Proposed model
Classes	COVID, No-Findings, Pneumonia	ILSVRC-2012 dataset 1000 classes	Multiple classes (car, person, bus etc.)	Multiple classes	Normal, pneumonia, and COVID-19	Normal, pneumonia viral, and COVID-19, pneumonia bacterial	COVID-19 and viral pneumonia
Model	Dark Covid Net	Conv Net	Mobile Net Conv Mobile Net	Inception-v3	Xception and ResNet50V2	Mobilenetv2 and VGG16	Inception ,Xception and ResNet50
Accuracy	87.00%	93.48%	94.72%	92.85%	92.85%	96.4%	98.03%
Easy to train	✗	✗	✗	✗	✗	✗	✓
Avoids overfitting	✗	✗	✗	✗	✗	✗	✓
Fast training time	✗	✗	✗	✗	✗	✗	✓
Easy to generalize	✗	✗	✗	✗	✗	✗	✓
Less no. of layers	✗	✗	✗	✗	✗	✗	✓
Avoids vanishing gradient	✗	✗	✗	✗	✗	✗	✓
Works on slow devices	✗	✗	✗	✗	✗	✗	✓
Includes regularization	✗	✗	✗	✗	✗	✗	✓
Easy to distribute	✗	✗	✗	✗	✗	✗	✓

Table 14. Comparative analysis of our proposed model with state of the art approaches.

Data availability

The X-ray of chest data is publicly available in <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database>.

Received: 28 August 2023; Accepted: 3 January 2024

Published online: 11 January 2024

References

- Sharma, S., Sharma, M. & Singh, G. A chaotic and stressed environment for 2019-nCoV suspected, infected and other people in India: Fear of mass destruction and causality. *Asian J. Psychiatry* **51**, 102049 (2020).
- Asghari, P., Rahmani, A. M. & Javadi, H. S. A medical monitoring scheme and health-medical service composition model in cloud-based IoT platform. *Trans. Emerg. Telecommun. Technol.* **30**(6), e3637. <https://doi.org/10.1002/ett.3637> (2019).
- Cao, C. *et al.* Deep learning and its applications in biomedicine. *Genom. Proteom. Bioinform.* **16**(1), 17–32 (2018).
- Kanji, J. N. *et al.* False negative rate of COVID 19 PCR testing: A discordant testing analysis. *Viol. J.* <https://doi.org/10.1186/s12985-021-01489-0> (2021).
- Ben Hassen, H., Dghais, W. & Hamdi, B. An E-health system for monitoring elderly health based on Internet of Things and Fog computing. *Health Inf. Sci. Syst.* **7**, 24. <https://doi.org/10.1007/s13755-019-0087-z> (2019).
- Aggarwal, S. *et al.* Automated COVID-19 detection in chest X-ray images using fine-tuned deep learning architectures. *Expert Syst.* **39**(3), e12749. <https://doi.org/10.1111/exsy.12749> (2022).
- Liu, L. *et al.* A smart dental health-IoT platform based on intelligent hardware, deep learning, and mobile terminal. *IEEE J. Biomed. Health Inform.* **24**(3), 898–906 (2020).
- Chollet, F. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1251–1258 (2017).
- Hassan, E., Talaat, F. M., Hassan, Z. & El-Rashidy, N. *Breast Cancer Detection: A Survey. Artificial Intelligence for Disease Diagnosis and Prognosis in Smart Healthcare* 169–176 (CRC Press, 2023).
- Maier, A., Syben, C., Lasser, T. & Riess, C. A gentle introduction to deep learning in medical image processing. *Z. Med. Phys.* **29**(2), 86–101 (2019).
- Guo, J. M., Yang, J. S., Seshathiri, S. & Wei, W. W. A light-weight CNN for object detection with sparse model and knowledge distillation. *Electronics (Switzerland)* **11**, 4 (2022).
- Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **60**(6), 84–90 (2017).
- Samee, N. A. *et al.* Metaheuristic optimization through deep learning classification of COVID-19 in chest X-ray images. *Comput. Mater. Continua* **73**, 2 (2022).
- Toğaçar, M., Ergen, B., Cömert, Z. & Özyurt, F. A deep feature learning model for pneumonia detection applying a combination of mRMR feature selection and machine learning models. *Irbm* **41**(4), 212–222 (2020).
- Hassan, E., Shams, M. Y., Hikal, N. A. & Elmougy, S. COVID-19 diagnosis-based deep learning approaches for COVIDx dataset: A preliminary survey. *Artif. Intell. Dis. Diagn. Progn. Smart Healthc.* **107**, 14 (2023).
- Gamel, S. A., Hassan, E., El-Rashidy, N. & Talaat, F. M. Exploring the effects of pandemics on transportation through correlations and deep learning techniques. *Multimed. Tools Appl.* **20**, 1–22 (2023).
- ElAraby, M. E., Elzeki, O. M., Shams, M. Y., Mahmoud, A. & Salem, H. A novel Gray-Scale spatial exploitation learning Net for COVID-19 by crawling Internet resources. *Biomed. Signal Process. Control* **73**, 103441 (2022).
- Laudicina, P. F. & Wean, D. Image enhancement techniques. *Semin. Radiol. Technol.* **9**(3), 108–119 (2001).
- Neary, P. L. Automatic hyperparameter tuning in deep convolutional neural networks using asynchronous reinforcement learning. In *Proceedings—2018 IEEE International Conference on Cognitive Computing, ICC3 2018—Part of the 2018 IEEE World Congress on Services (July 2018)*, 73–77 (2018).
- Sharma, A., Singh, K. & Koundal, D. A novel fusion based convolutional neural network approach for classification of COVID 19 from chest X-ray images. *Biomed. Signal Process. Control* **77**(January), 103778. <https://doi.org/10.1016/j.bspc.2022.103778> (2022).

21. Uddin, A., Bayazid, T., Mohammad, M. K., & Atef, Z. Study on convolutional neural network to detect COVID 19 from chest X-rays (2021).
22. Science, Computer, and South Florida. Cooking Object's State Identification Without Using Pretrained Model.
23. Majeed, T., Rashid, R., Ali, D. & Asaad, A. COVID 19 detection using CNN transfer learning from X-ray images persistent homology for image analysis view project COVID 19 detection using CNN transfer learning from X-ray images, 1–10. <https://www.researchgate.net/publication/341511596> (2020).
24. Naveena, S. & Ayyasamy, B. A new design of diabetes detection and glucose level prediction using moth flame-based crow search deep learning. *Biomed. Signal Process. Control* **77**(2021), 103748. <https://doi.org/10.1016/j.bspc.2022.103748> (2022).
25. Aggarwal, S. *et al.* Automated COVID 19 detection in chest X-ray images using fine-tuned deep learning architectures. *Expert Syst.* **39**(3), 1–17 (2022).
26. Das, D. & Santosh Umapada, K. C. Truncated inception net: COVID-19 outbreak screening using chest X-rays. *Phys. Eng. Sci. Med.* **43**(3), 915–925. <https://doi.org/10.1007/s13246-020-00888-x> (2020).
27. Hou, J. & Terry, G. Explainable DCNN based chest X-ray image analysis and classification for COVID-19 pneumonia detection. *Sci. Rep.* <https://doi.org/10.1038/s41598-021-95680-6> (2021).
28. Sanket, S., Vergin Raja Sarobin, M. & Jani Anbarasi, L. Detection of novel coronavirus from chest X-rays using deep convolutional neural networks. *Multimed. Tools Appl.* <https://doi.org/10.1007/s11042-021-11257-5> (2022).
29. Litjens, G. *et al.* A survey on deep learning in medical image analysis. *Med. Image Anal.* **42**, 60–88 (2017).
30. Kermary, D. S. *et al.* Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell* **172**(5), 1122–1131. e9 (2018).
31. Jaiswal, A. K. *et al.* Identifying pneumonia in chest X-rays: A deep learning approach. *Meas. J. Int. Meas. Confed.* **145**, 511–518 (2019).
32. Singh, D., Kumar, V., Vaishali, & Kaur, M. Classification of COVID 19 patients from chest CT images using multi-objective differential evolution-based convolutional neural networks. *Eur. J. Clin. Microbiol. Infect. Dis.* **39**(7), 1379–1389 (2020).
33. Narayan Das, N., Kumar, N., Kaur, M., Kumar, V. & Singh, D. Automated deep transfer learning-based approach for detection of COVID 19 infection in chest X-rays. *IRBM* <https://doi.org/10.1016/j.irbm.2020.07.001> (2020).
34. Brunese, L., Mercaldo, F., Reginelli, A. & Santone, A. Explainable deep learning for pulmonary disease and coronavirus COVID 19 detection from X-rays. *Comput. Methods Programs Biomed.* **196**, 105608 (2020).
35. Jain, G., Mittal, D., Thakur, D. & Mittal, M. K. A deep learning approach to detect COVID 19 coronavirus with X-ray images. *Biocybern. Biomed. Eng.* **40**(4), 1391–1405 (2020).
36. Varela-Santos, S. & Melin, P. A new approach for classifying coronavirus COVID 19 based on its manifestation on chest X-rays using texture features and neural networks. *Inf. Sci.* **545**, 403–414 (2021).
37. Marques, G., Agarwal, D. & de la Torre Diez, I. Automated medical diagnosis of COVID 19 through EfficientNet convolutional neural network. *Appl. Soft Comput.* **96**, 106691 (2020).
38. Ezzat, D., Hassanien, A. E. & Ella, H. A. An optimized deep learning architecture for the diagnosis of COVID 19 disease based on gravitational search optimization. *Appl. Soft Comput.* **98**, 106742 (2021).
39. Hassantabar, S., Ahmadi, M. & Sharifi, A. Diagnosis and detection of infected tissue of COVID 19 patients based on lung X-ray image using convolutional neural network approaches. *Chaos Solitons Fractals* **140**, 110170 (2020).
40. Elzeki, O. M., Shams, M., Sarhan, S., Abd Elfattah, M. & Hassanien, A. E. COVID-19: A new deep learning computer-aided model for classification. *PeerJ Comput. Sci.* **7**, e358 (2021).
41. Shibly, K. H., Dey, S. K., Islam, M. T. U. & Rahman, M. M. COVID faster R-CNN: A novel framework to diagnose novel coronavirus disease (COVID-19) in X-ray images. *Inform. Med. Unlocked* **20**, 100405 (2020).
42. Reshi, A. A. *et al.* An efficient CNN model for COVID-19 disease detection based on X-ray image classification. *Complexity* **2021**, 1–12 (2021).
43. Pham, T. D. Classification of COVID-19 chest X-rays with deep learning: New models or fine tuning?. *Health Inf. Sci. Syst.* **9**, 1–11 (2021).
44. Elzeki, O. M., Abd Elfattah, M., Salem, H., Hassanien, A. E. & Shams, M. A novel perceptual two layer image fusion using deep learning for imbalanced COVID-19 dataset. *PeerJ Comput. Sci.* **7**, e364 (2021).
45. Elmuog, S., Hikal, N. A. & Hassan, E. An efficient technique for CT scan images classification of COVID-19. *J. Intell. Fuzzy Syst.* **40**(3), 5225–5238 (2021).
46. Wahab, M. N. *et al.* Efficientnet-lite and hybrid CNN-KNN implementation for facial expression recognition on raspberry Pi. *IEEE Access* **9**, 134065–134080 (2021).
47. Patel, R., & Anita, C. Transfer learning with fine-tuned MobileNetV2 for diabetic retinopathy. In *2020 International Conference for Emerging Technology, INCET 2020*, 6–9 (2020).
48. Heidari, M. *et al.* Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms. *Int. J. Med. Inform.* **144**, 104284 (2020).
49. Rahman, T., Chowdhury, M., & Khandakar, A. COVID 19 Radiography Database. Kaggle, version 4. <https://www.kaggle.com/tawsifurrahman/covid19-radiography-database> (2021).
50. Chowdhury, M. E. H. *et al.* Can AI help in screening viral and COVID 19 pneumonia?. *IEEE Access* **8**, 132665–132676 (2020).
51. Soomro, T. A. *et al.* Impact of image enhancement technique on CNN model for retinal blood vessels segmentation. *IEEE Access* **7**, 158183–158197 (2019).
52. Rubini, C. & Pavithra, N. Contrast enhancement of MRI images using AHE and CLAHE techniques. *Int. J. Innov. Technol. Explor. Eng.* **9**(2), 2442–2445 (2019).
53. Vaddi, R. S., Boggavarapu, L. N.P., Vankayalapati, H. D. & Anne, K. R. Comparative analysis of contrast enhancement techniques between histogram equalization and CNN. In *3rd International Conference on Advanced Computing, ICoAC 2011*, 106–110 (2011).
54. Shijie, J., Wang, P., Jia, P., & Hu, S. Research on data augmentation for image classification based on convolution neural networks. In *Proceedings—2017 Chinese Automation Congress, CAC 2017* 2017–January(201602118), 4165–70 (2017).
55. Jiang, X. *et al.* CapsNet, CNN, FCN: Comparative performance evaluation for image classification. *Int. J. Mach. Learn. Comput.* **9**, 6 (2019).
56. Ozturk, T. *et al.* Automated detection of COVID 19 cases using deep neural networks with X-ray images. *Comput. Biol. Med.* **121**, 103792 (2020).
57. Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. [arXiv:1704.04861](https://arxiv.org/abs/1704.04861) (arXiv preprint) (2017).
58. Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, California, USA*. <https://ojs.aaai.org/index.php/AAAI/article/view/11231> (2017).
59. Rahimzadeh, M. & Attar, A. A modified deep convolutional neural network for detecting COVID 19 and pneumonia from chest X-ray images based on the concatenation of xception and resnet50v2. *Inform. Med. Unlocked* **19**, 100360. <https://doi.org/10.1016/j.imu.2020.100360> (2020).

Acknowledgements

Thank you to all of the authors who made significant contributions to this paper.

Author contributions

B.P.K.: draft writing, data collection, implementation, formatting; H.S.: mentoring, writing, data collection; R.H.: mentoring, data collection, formatting; S.K.S.: data collection; C.K.: data collection; M.M.H.: formal analysis, project administration, final validation, writing; M.A.S. mentoring, funding, writing.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to M.A.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2024