COVID-19 Cases Detection from Chest X-Ray Images using CNN based Deep Learning Model

Md Amirul Islam[1], Giovanni Stea[2], Sultan Mahmud[3], Kh. Mustafizur Rahman[4]
Department of Information Engineering, University of Pisa, Italy[1,2,3]
Faculty of Science & Technology, Bangladesh University of Professionals, Bangladesh[4]

***Abstract*—COVID-19 has recently manifested as one of the most serious life-threatening infections and is still circulating globally. COVID-19 can be contained to a considerable extent if a patient can know their COVID-19 infection at a possible earlier time, and they can be isolated from other individuals. Recently, researchers have explored AI (Artificial Intelligence) based technologies like deep learning and machine learning strategies to identify COVID-19 infection. Individuals can detect COVID-19 disease using their phones or computers, dispensing with the need for clinical specimens or visits to a diagnostic center. This can significantly reduce the risk of spreading COVID-19 farther from a probably infected patient. Motivated by the above, we propose a deep-learning model using CNN (Convolutional Neural Networks) to autonomously diagnose COVID-19 disease from CXR (Chest X-ray) images. The dataset used to train our model includes 10293 X-ray images, with 875 X-ray images from COVID-19 cases. The dataset contains three different classes of the tuple: COVID-19, pneumonia, and normal cases. The empirical outcomes show that the proposed model achieved 97% specificity, 96.3% accuracy, 96% precision, 96% sensitivity, and 96% F1-score, respectively, which are better than the available works, despite using a CNN with fewer layers than those.**

Keywords*—COVID-19; CNN; deep learning; machine learning; chest X-ray

I. Introduction

Wuhan, a business hub in China's Hubei province, saw a fresh coronavirus outbreak in 2019. Researchers in China termed the novel virus the 2019 n-Cov or the Wuhan virus [1]. Officially a study group found this virus, and they named it SARS-CoV-2. During the coronavirus infection outbreak in 2019, the international committee was labeled coronavirus disease 19 (COVID-19) [2, 3, 4]. In the first place, coronaviruses have caused disease in people who have been exposed to wild animals, primarily bats and rats [5, 6, 7]. After almost three years since its first appearance, COVID-19 shows no signs of abating, and has already infected half a billion individuals and claimed more than six million deaths.

COVID-19 is a disease caused by a virus source that aggravates the lungs, causing pneumonia in patients. However, the treatment and medication of these cases are quite different from pneumonia cases arising from other viruses or bacteria. If an individual shows symptoms of COVID-19, certain precautionary steps are taken in conjunction with the diagnosis. COVID-19 patient is isolated for a certain number of days in order to contain the further spread of the infection. Therefore, to stop the spread of COVID-19, accurate and timely identification of pneumonia caused by the virus is a critical issue.

The WHO approved a testing method based on Polymerase chain reaction (RT-PCR), whereby short DNA or RNA sequences are analyzed and replicated or intensified [8]. In some circumstances, more than one test may be required to rule out the possibility of coronavirus infection. According to the WHO laboratory research, negative results do not always mean that a person is not infected with COVID-19 [9].

Although RT-PCR examination is the most trustworthy technique to diagnose COVID [10], it is a laborious, time consuming, complex manual procedure, with kits in short supply depending on the time and place. Furthermore, the test is unpleasant and slightly invasive, as it entails collecting nasopharyngeal swabs. COVID-19 attacks the human airway epithelial (hAE) cells. Accordingly, X-rays can be used as a potential specimen to know the damaged portions of a COVID-19 patient's lungs. Detection of COVID-19 using X-ray images of a patient might be a helpful technique due to its fast speed, contained cost, and wide range of applications [11].

The insufficient supply of COVID-19 scanning workstations and research kits poses challenges for the medical practitioners and personnel to cope up with COVID-19. Rapid and effective identification of suspicious COVID-19 cases is of fundamental importance to contain the spread of the infection. The need for repeated checks to ascertain the actual condition and make effective decisions is often exponential growth in critical situations for patients. Therefore, we set the following objectives to design a machine-learning based COVID-19 case detection system.

We summarize the objectives as follows:

- Assisting radiologists and medical specialists in detecting subtle and gradual changes in X-rays that might otherwise go unnoticed;
- Many people in developing countries do not have access to radiologists due to high costs. This tool could help them read their X-ray images to classify them as COVID-19, pneumonia, and normal;
- Creating a model to scan complex data such as CT and MRI images to detect COVID-19 cases.

A deep-learning approach, such as CNNs, is able to learn and deduce properties of a very complex nonlinear functions autonomously, without the need for human intervention. CNNs act exclusively on input data after a supervised training phase. For example, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) reported a well-known scenario when a

model outperformed humans in the task of classifying images in 2015 [12].

The primary purpose of this study is to build a deep-learning paradigm to diagnose pneumonia from CXR images as well as the location and positioning of abnormalities in X-rays. In addition, we propose a deep-learning-based model to detect cases of COVID-19, influenza, and normal X-ray picture classification. This model provides a low-cost method for radiologists and medical professionals to cross-check their interpretations and recognize any possible results that might otherwise have been overlooked. The model is trained with the dataset collected from Kaggle and GitHub. The radiological society and researchers recently formed these datasets from patients' X-ray images during the COVID-19 outbreak. To improve the model's accuracy, we adopt clinical image engineering that has consolidated imaging computing approaches such as pre-processing techniques, classification approaches, and illness screening and localization in a sequential manner.

Following are the contributions of this paper:

- We developed a CNN-based architecture to detect COVID-19 from patients' X-rays after collecting and pre-processing X-ray data. We trained our model by using Keras on top of TensorFlow. The proposed approach achieves 96.3% accuracy for the Pneumonia, Normal, and COVID-19 classes. Despite our model being less complex (as measured by the number of layers in the CNN) than the available ones, it outperforms them all in accuracy.
- Grad-CAM has been employed to detect characteristic features from X-ray images to aid visually interpretative decision-making about COVID-19.

The rest of this paper is summarized as follows: In Section II, recent research works are represented. Our suggested approach and material with CNN-based architecture for the COVID-19 detection system are discussed in Section III. Experimental results and discussion are illustrated in Section IV and Section V concludes our paper and highlights directions for future work. Table I reports a list of the acronyms used in this paper.

II. Literature Review

Researchers have been utilizing different deep-learning strategies to identify COVID-19 from clinical images such as X-rays and CT scans of the chest. Rahimzadeh et al. [13] built a joint CNN-assisted ResNet50V2 and Xception model for categorizing COVID-19 cases by employing CXR images. The dataset fed into this method included 8851 images of healthy people, 6054 images of pneumonia patients, and 80 images of COVID-19 patients. In this model, 633 images were chosen for each of the eight training phases. For COVID-19 cases, this system achieved an experimental result of 99.56% accuracy and 80.53% recall.

Alqudah et al. [14] utilized AI strategies to implement a method for detecting COVID-19 from CXR images. Several machine-learning strategies like Support Vector Machine (SVM) and Random Forest (RF) were picked for classification purposes. The system achieved 95.2% accuracy, 93.3% sensitivity, and 100% specificity. GAN with deep learning to identify the COVID-19 from CXR images was suggested by Loey et al. [7]. GoogleNet, AlexNet, and RestNet18 are the pre-trained models used by this proposed framework for identifying COVID-19 cases. The GoogleNet pre-trained model was picked as a primary deep-learning technique. The model showed test accuracy of 80.6% within the four classes continuity (79% normal cases, 79% bacterial pneumonia cases, 79% images of pneumonia cases, and 69% COVID-19 cases). In the same way, for three classes continuity, AlexNet gained test accuracy 85.2%, and GoogleNet attained test accuracy of 99.9% for the two classes continuity.

Kumar et al. [15] suggested a deep-learning procedure to identify and classify COVID-19 infected patients. The proposed approach was trained by using nine pre-trained processes to extract features from the CXR images, and for the classification they used SVM. The datasets included 158 CXR images of both COVID-19 and non-COVID-19 cases. Obtaining 95.52% F1-score and 95.38% accuracy, ResNet50 combined with the SVM method was a statistically outstanding technique.

Horry et al. [16] suggested a COVID-19 identifying framework from CXRs considering pre-trained methods. The recommended framework applied Inception, ResNet, Xception, and VGG to classify COVID-19 cases. The dataset employed within the framework encompasses 60361 images of normal cases, 322 images of pneumonia cases, and 115 images of COVID-19 cases. They were found recall and precision values 80% for the VGG19 and VGG16 classifiers.

Ucar et al. [17] introduced a COVID-19 identifying framework by utilizing X-ray photos and deep learning techniques. The data set used for the framework contained 1583 photos of normal cases, 76 photos of COVID-19 cases, and 4290 photos of pneumonia cases, and for COVID-19 cases, the system attained 98.3% accuracy. A transfer-learning technique with the CNN is recommended by Apostolopoulos et al. [18]. By extracting essential attributes from CXR images, this technique could instinctively diagnose COVID-19 patients. This recommended framework used five CNN models, including InceptionResNetV2, MoblieNet, Inception, Xception, and VGG19, to sort the COVID-19 CXR images. The dataset for three classes contained 504 CXR images of normal cases, 700 CXR images of pneumonia cases, and 224 CXR images of COVID-19 cases. VGG19 was chosen as the primary method with 98.75% sensitivity, 92.85% specificity, and 93.48% accuracy within the introduced framework.

A complete unique model introduced by Bandyopadhyay et al. [19] that made use of the LSTM-GRU to categorize confirmed automatically, negative, positive, recovered, and death cases of coronavirus. The above framework attained 67.8% accuracy for negative cases, 62% accuracy for death cases, 40.5% accuracy for released cases, and confirmed cases with 87% accuracy.

A deep-learning network to identify COVID-19 cases autonomously from CXRs was presented by Khan et al. [20]. The dataset includes 310 images of normal cases, 330 images of bacterial pneumonia cases, 327 images of the viral pneumonia cases, and 284 images of COVID-19 cases. With COVID-19, the recommended model acquired 93.5% accuracy, 100% recall, and 97% precision. A complete unique framework

TABLE I. LIST OF ACRONYMS USED IN THE PAPER

Notation	Definition
AI	Artificial intelligence
AP	Attending Physician
AUC	Area Under the Curve
COVID-19	Coronavirus
CNN	Convolutional Neural Network
CT	Computed Tomography
CXR	Chest X-ray
CMC	Composite Monte-Carlo
CSL	Cost Senseitive Learning
DNN	Deep Neural Network
FCL	Fully Connected Layers
FP	False Positive
FN	False Negative
Grad-CAM	Gradient Class Activation Map
GAN	Generative Adversarial Network
Grad-CAM	Gradient-Weighted Classes Activation Mapping
GPU	Graphical Processing Unit
ILSVRC	ImageNet Large-Scale Visual Recognition Challenge
KTD	Knowledge Transfer and Distribution
LR	Learning Rate
MRI	Magnetic Resonance Imaging
PCA	Principal Component Analysis
RT-PCR	Reverse Transcription Polymerase Chain Reaction
RF	Random Forest
ReLU	Rectified Linear Unit
ROC	Receiver Operating Characteristic Curve
SARS-CoV-2	CoronaVirus-2 Severe Acute Respiratory Syndrome
SVM	Support Vector Machine
TP	True Positive
TN	True Negative
WHO	World Health Organisation
2019 n-Cov	2019-Novel-Coronavirus

based on a deep-learning approach to identify COVID-19 from CXR images called COVIDX-Net was represented by Hemdan et al. [21]. Authors used eight pre-trained models contained Xception, InceptionResNetV2, InceptionV3, VGG19, DenseNet201, MobileNetV2, and ResNetV2 for their frameworks. The dataset contained 50 CXR images, with 25 images of non-COVID-19 cases and 25 of COVID-19 cases. VGG19 and DenseNet201 achieved 83% precision and 90% accuracy.

In [22], Singh et al. utilized a VGG16 model as a deep transfer-learning framework to identify COVID-19 patients from CT scan images. In this proposed framework, authors used four classifiers models to classify COVID-19 cases, and the PCA extracted the features from CXR images. With the help of the SVM classifier and bagging ensemble method, 95.7% accuracy, 95.3% F1-score, and 95.8% precision achieved effective result.

In [23] the authors presented a lightweight DNN-based mobile app named COVID-MobileXpert, that can use noisy photos of CXRs for point-of-care COVID-19 screening. In addition, they created a new 3-players Knowledge Transfer and Distribution (KTD) framework that includes a pre-trained attending physician network that extracted photo attributes from a gigantic set of lung infection CXR photos. Furthermore, a fine-tuned RF network retains the fundamental CXR imaging attributes to distinguish COVID-19 cases from normal and pneumonia CXR images.

To detect COVID-19 symptoms, Ahuja et al. [24] suggested a transference learning strategy employing a three-phase approach. With a view to indicating the abnormality of the image, several pre-trained models had been applied with the augmented image using ResNet18 deep-learning model, and thus, 99.4% of test cases accuracy were obtained. In [25] the authors talked about deep learning emerged case study utilizing CMC & fuzzy rule induction to deal with the delimited information for forecasting strategies. To identify COVID-19 cases, Ashraf et al. [26] presented a framework called COVID-CAPS. In [27] authors introduced a deep-learning algorithm, which can perform fast detection of COVID-19 cases, and their system attained 95% accuracy. Karthik et al. [28] suggested a custom CNN-based architecture to detect COVID-19 cases, and for each kind of pneumonia, it can learn unique convolutional filter patterns.

TABLE II. NUMBER OF CNN LAYERS IN THE REVIEWED MODELS

Year	Ref.	Number of Layers
2020	[13]	6
2020	[14]	8
2020	[15]	≥ 1000
2020	[16]	≥ 200
2020	[17]	5×15 Layers/Sublayers
2020	[18]	≥ 1000
2020	[20]	74
2020	[21]	≥ 400
2020	[22]	2500 Hidden Layers with 8 Different Layers
2020	[23]	9
2021	[24]	71
2020	[26]	7
2020	[27]	27×19 Layers/Sublayers
2021	[28]	10
2022	**Proposed Model**	**5**

Aside from that, researchers continually contribute to inventing feasible methods to identify COVID-19 cases from CXR images. The above review establishes that deep learning, especially CNN, plays an important role in COVID-19 diseases

detection and classification in medical imaging. However, the recent study has limitations in identifying the COVID-19 virus in normal cases from the CXR images. Our proposed model is novel and different from the others presented above. It has fewer layers, which entails smaller complexity. The number of layers used in the methods reviewed in this section is shown in Table II, and is always larger, and often considerably so. As we will show later, despite being less complex, our model achieves better performance in most metrics.

III. Proposed COVID-19 Detection System

This section presents the primary contribution of the paper: the architectural design and implementation of the suggested methodology.

A. Methodology

The architecture of our system for COVID-19 detection consists of several building blocks as illustrated in Fig. 1.

Initially, raw CXR images are provided in the pre-processing pipeline for performing pre-processing tasks like resizing, normalization, flipping, zooming, and rotation. After the pre-processing phase, the data set is divided into a training and a test set. Next, the proposed system is trained using the training data. Training and validation accuracy and loss are determined after every epoch. The following evaluation metrics have been adopted for performance measurement: accuracy, sensitivity, specificity, F1-score, precision, Area Under the Curve (AUC) using Receiver Operating Characteristics (ROC) and confusion matrix.

1) Collection of Dataset and Explanation: As COVID-19 has recently broken out, datasets with more sample X-rays tagged with COVID-19 cases are not available yet. As a result, collecting data on various image sources of normal, COVID-19, and pneumonia cases need to be collected. For COVID-19 cases, 2875 CXR images have been collected from GitHub [29] and Kaggle [30]. For pneumonia and normal cases, 4200 and 3218 CXR images are collected from the Kaggle repository [30, 31]. In total, our dataset contains 10293 CXR images. Later, the images are resized to a resolution of 224x224 pixels. Fig. 2 depicts the visualization of several CXR images of every class.

The number of CXR images of each set was partitioned in Table III

TABLE III. Dataset

Normal	Covid	Pneumonia	Total
3218	2875	4200	10293

2) Image Pre-Processing: To improve the model's efficiency, pre-processing of the image is required. In this work, we also carried out certain pre-processing activities to produce better performance. The pre-processing techniques used in this research are listed in this section.

A.1 Resize Picture to Capture the Central Portion and Remove Black Bars: Typically, square images are required to train CNN models. The CXR data obtained from various sources differs in sizes. So, in order to feed CXR images into the model, we need to scale the images down to a square shape. However, this causes an asymmetry of the images as seen in the Fig. 3.

To tackle the distortion of the input images, we applied Pasa et al. [32]'s technique to remove the central area and delete black bars. Fig. 4 depicts a pre-processed image obtained by carrying out the following operations:

1) If any black bands appear at the image's margins, they are discarded;
2) The image's dimension is warped until the minimum boundary counts 224 pixels;
3) Retrieve the 224×224 pixel core area.

A.2 Normalization: Subsequently, images are normalized and transformed to the proper data type. Original images are grayscale with individual pixels coded as Unit8 type with values in range 0-255. For the Keras model, data is provided in float32 type. Therefore, the pre-processed images have to be converted into float32, normalized to a range $[0; 1]$. To normalize every pixel, the data value is divided by the highest value of uint8, which is 255.

A.3 Data Augmentation: Deep-learning methods, such as CNN, generates more accurate outcomes if larger numbers of full-sized images can be used. As a result, data augmentation is useful in creating the first training images that are backed up by extra data. We conducted the subsequent data augmentation procedures during training:

- 5 to 10 degrees of spontaneous rotation;
- Zooming within a range of +10% and -10%;
- Flipping horizontally.

3) Proposed CNN Model Development: A deep-learning model has been built in this paper to diagnose COVID-19 incidents. The model is trained using a dataset having three types of CXR images. The suggested CNN model structure for COVID-19 detection is shown in Fig. 5.

The proposed CNN model has five convolutional blocks. Each convolutional block has multiple layers, and each layer has one activation function named Rectified Linear Unit (ReLU). The third block and fourth block have a dropout layer to reduce the over-fitting problem. Two fully connected layers (FCLs) have been used, the first FCL is used with the dropout layer, and the last FCL is connected with the softmax classifier.

A.1 Convolution Layer: Every convolution block of this model has multiple convolution layers. The first convolution block consists of conv1-layer1, and conv1-layer2. Similarly, second convolution block has conv2-layer1, and conv2-layer2. Third convolution block has conv3-layer1, conv3-layer2. Fourth convolution block consists of conv4-layer1, and conv4-layer2, and final fifth block has conv5-layer1, and conv5-layer2. The fourth block and fifth block have an extra dropout layer to reduce the overfitting problem. Conv1 block has used a total of 32 filters where every filter size is 5 x 5. Conv2 block uses a total of 64 filters where every filter size is 3 x 3. Conv3 block uses a total of 128 filters where every filter size is 3 x 3. Conv4 block uses a total of 256 filters where every filter size is 3 x 3, and similarly, Conv5 block has used a total of 512 filters where every filter size is 3 x 3.

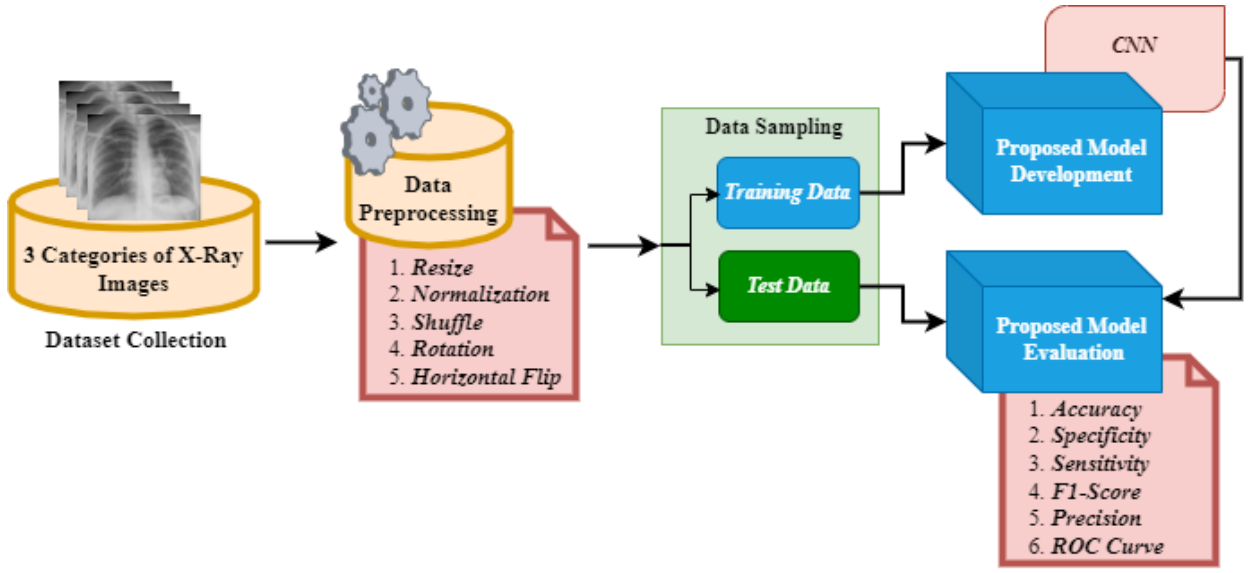


Fig. 1. Proposed Architecture for COVID-19 Detection System.

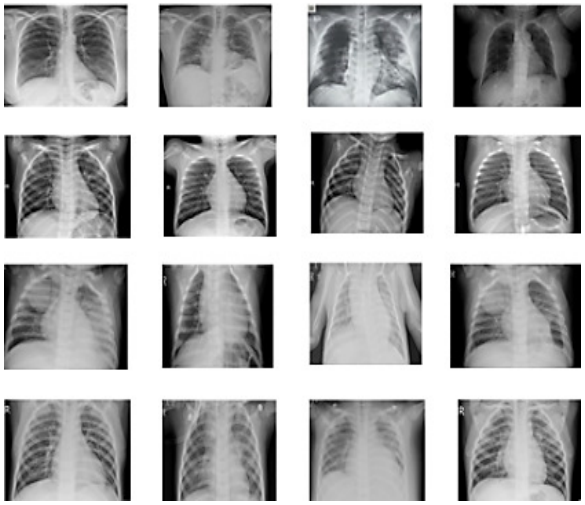


Fig. 2. X-ray Images

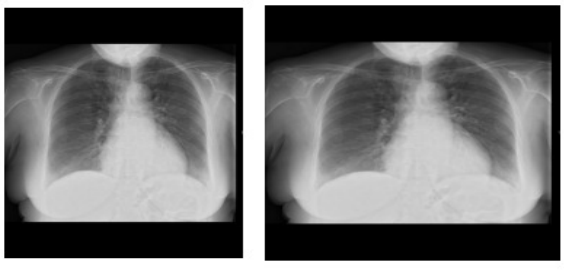


Fig. 3. Distortion Due to the Images being Resized to a Square Shape. (a) Displays the Original CXR Image of a COVID Patient; (b) Displays the Resampled Image in a Square Shape.

The functional process performed by the convolutional layer

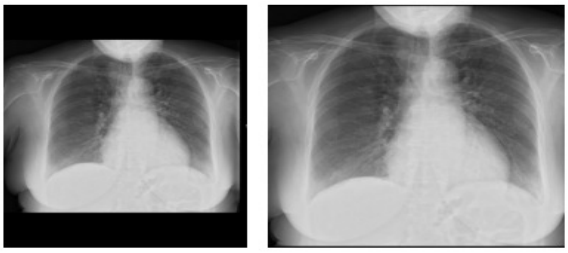


Fig. 4. Pre-Processing is Applied to All Dataset Files. (a) Displays the Original CXR Image of an Individual with COVID-19 (b) Displays a Square Shape of the Pre-Processed Image.

is:

$$F(i,j) = (J \times K)(i,j) = \sum\sum J(i+p, j+q)K(p,q), \quad (1)$$

where the input matrix is represented with J, a 2D filter of size $p \times q$ is indicated by K, and F denotes a 2D feature map's output. $J \times K$ describes the operation of the convolutional layer.

A.2 Rectified Linear Unit: Each convolution layer has an activation function. Here this model used the ReLU activation function. To extend nonlinearity in feature maps, the ReLU layer is utilized [33]. ReLU itemizes activation by maintaining the threshold input at zero. It is mathematically represented as follows:

$$f(y) = max(0, y) \quad (2)$$

A.3 Zero Padding Layer: The application of this layer is to add columns and rows of zeros to the left, right, top, and

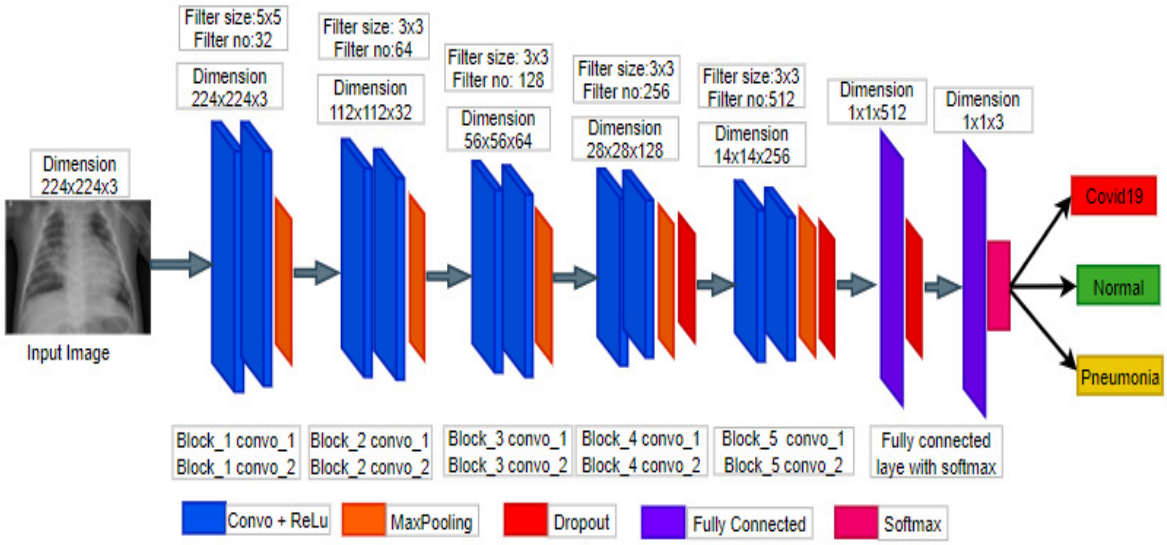


Fig. 5. Proposed CNN Architecture.

bottom sides of an image. We have applied 1×1 zero padding during our work.

A.4 Pooling Layers : The next layer that we applied is a pooling layer. This layer is used to reduce the number of consumption and parameters in the network by dynamically shrinking the spatial size of the images. The pooling layer aids in reducing the problem of overfitting and executes distinct operations on each depth slice of the input and resizes it spatially using the MAX function. The most common MAX pooling layer with filters of size 2×2 is used with a stride of 2. MAX pooling is applied to downsample the input by two along width and height, discarding 75% of the activation. The pooling layer is represented below:

1) Accepted a volume of size is $w1 \times h1 \times d1$
2) Required two hyper parameters:
   - F is their spatial extent
   - S is the stride
3) Produced a volume of size is $w2 \times h2 \times d2$:
   - $w2 = (w1 \times F)/S + 1$
   - $h2 = (h1 \times F)/S + 1$
   - $d2 = d1$
4) It presents zero parameters since it calculates a specified function of the input.
5) Zero padding is not commonly used in the pooling layers [34].

A.5 Dropout Layer : There are many methods to handle a CNN's capacity to eliminate overfitting. For example, the dropout layer [35] is an effective regularization method to prevent overfitting problems. During training time, the dropout layer randomly sets some neuron's activation to zero and those neurons will not update their weights. Dropout is activated with some probability during training, otherwise it is inactive. Some neurons do not learn all characteristics due to dropout. There is no dropout layer added throughout the testing period. Most of the times, a dropout ratio $p = 0.5$ shows good results, hence is selected as default. The p value can be tuned during data validation as well.

A.6 Fully Connected Layer: Each neuron in one layer is connected to another neuron in another layer, hence FCLs are formed. The layer basically takes an input image or object and outputs an N-dimensional vector as a response, where N is the number of given classes that the model or program has to choose from [36]. The working technique is as follows: the system looks at the output of the preceding layer and identifies which features are most commonly associated with a specific class. Fundamentally, a FCL considers which high-level features are most closely associated with a certain class and assigns weights to them, resulting in the classification value when the products of the weights and the preceding layer output are computed.

A.7 Softmax : The softmax function with loss [37] is used, which crushes an N-dimensional vector x of random real values to an N-dimensional vector (x) of real values in the range from 0 to 1 that will sum up to 1. The function is given by the following equation 3:

$$\sigma(x)_j = \frac{e_j^z}{\sum_{k=1}^{k} e_j^z} \tag{3}$$

Here, $j = 1, 2, ..., k$. For this model, the output feature map will be for three classes. In the output feature map, the pixels belonging to the predicted class will be 1, and for that same pixel, other classes will contain zero.

4) Cost-Sensitive Learning: Our proposed method focuses on learning features automatically from CXR images based on CNN. Cost-sensitive learning (CSL) [38] is a mechanism in which the method that ranks each class in a classification problem can be penalized. When using CNNs, CSL helps prevent prejudices in classification and then assists in overcoming the issue of imbalance as a tool. We have chosen CNN as our data set is imbalanced. The approach of class weight is used in this research as one possible way to mitigate the future effects of data imbalance. We adjust the weights in the class weight system in inverse proportion to the number of occurrences of each class in the raw data. We have used a Sci-Kit Learn feature for all this, which acquires numbers to match the number of cases centred on logistic regression concepts [39][40]. The weight W_k in class k is calculated using the following equation 4.

$$W_k = \frac{\sum Cases}{\sum Classes \times \sum Cases\ in\ class(k)} \tag{4}$$

The weights of the groups are utilized to match the standard. As a result, in the loss function, we assign higher values to cases involving smaller groups. The estimated loss would then be a weighted average, with W_k denoting the weights assigned to each class inside each sample throughout the loss calculation.

5) Loss Function Used: Categorical Cross-Entropy: The objective of network training is to maximize the probability of correct classification. This is gained by minimizing the cross-entropy loss for each training sample. The loss function utilized in our work is the categorical cross-entropy loss. Our cases are divided into three categories. The failure is determined independently for each class, and the values are then combined together. In the equation 5 [41] below, categorical cross-entropy is specified.

$$L(p, q) = -\sum W_k \times p_k \times log(q_k), \tag{5}$$

where k is the class number (in this work, the classes are COVID-19, Normal, and Pneumonia), q is the predicted probability or softmax function of class k and W_k is the weight for class k.

6) Screening and Localizing Pathogens: In order to better understand the stimulation of the last convolutionary layer of the developed model, we have applied the Gradient-Weighted Classes Activation Mapping (Grad-CAM) [42] algorithm. This layer is the only one that delivers the parameters for the logistic layer of the final parameters to determine a probability distribution output. For constructing the Grad-CAM of an input file, the gradients of this layer have been used. Grad-CAM thus gives a coarse localization map that shows the most significant areas in the picture (radiological features).

IV. Experimental Results and Discussion

In this section, we discussed the implementation phase and evaluated the classification efficiency of the proposed architecture. The performances and the characteristics of our proposed system are also compared with existing works for COVID-19 detection.

A. Experimental Process

The experiment performed in this research are described in this section.

1) Experimental Environment: The proposed approach uses a 0.001 learning rate and a 25-epoch number. On the 1.80 GHz Intel (R) Core i5-8265U Processor, the proposed CNN was developed utilizing $Keras$ and the $Python$ package with the $TensorFlow2$ backend. Additionally, the experiments were conducted utilizing Google Colab's graphical processing unit (GPU).

2) Split Data into Training set and Testing set: We divide the data into training data, consisting of 90% (9264 images) and testing that makes 10% (1029 images). For validation, we further divide the training data again as 10%. The images were distributed randomly across the test and training datasets, ensuring that the two datasets are shared roughly evenly between the two groups. It is also crucial that the model is trained in a variety of ways. To put it in another way, it would be overfitted, which is not a good thing.

3) Model Training Methodology: The Adam optimizer was used to train every layer of the improvement parcel with the normal performance parameters ($\beta 1 = 0.9$ and $\beta 2 = 0.999$) using batches of size 32 and running throughout every epoch through the whole dataset. Eventually, adding an effective learning rate scheduler called "ReduceLROnPlateau" defines a lookup to control the relevance of the loss of reliability. It starts with a learning rate of 0.001 on demand.

We reduced the hyperparameter learning rate by a factor of 0.5 after 5 epochs, with no improvement recorded by the detector. We have also included an early stop to prevent the network from over-learning in the regularization step. We stopped training the model after 5 epochs because the loss score had not changed. The hyperparameters used in the proposed model are shown in Table IV.

B. Performance Evaluation Metrics

Five metrics, i.e. accuracy, sensitivity, specificity, F1-score, and precision, are used to assess the proposed system's performance. Some variables are employed in the metrics calculation, as mentioned below:

- True Positives (TPs) are the COVID-19 cases correctly estimated;
- False Positives (FPs) are pneumonia or normal cases that are misclassified as COVID-19;
- True Negatives (TNs) are correctly classified pneumonia or normal cases;
- False Negatives (FNs) aer COVID-19 cases that are misclassified as pneumonia or normal cases.

Using the above mentioned variables, performance evaluation metrics can be computed as:

- Accuracy: A metric that normally expresses how the model performs across all classes. It is determined as the proportion between the quantity of right prediction to the all the number of predictions, i.e.:

TABLE IV. HYPER PARAMETERS OF THE NETWORK

Hyper Parameter	Weight
Batch size	32
Cost function	Categorical Cross Entropy
Learning Rate (LR)	0.001
LR Multiplying factor	0.5
LR Decay	5 times after a plateau
Epochs	25
Optimizer	Adam

$$Accuracy = \frac{(TN + TP)}{(TP + FP + TN + FN)} \quad (6)$$

- Sensitivity: It indicates the ratio of individuals who test positive among all those who have the diseases, expressed as:

$$Sensitivity = \frac{TP}{(TP + FN)} \quad (7)$$

- Specificity: The ratio of individuals who test negative among all those that actually do not have that disease, expressed as:

$$Specificity = \frac{TN}{(TN + FP)} \quad (8)$$

- F1-score: The F1-score is a count of a test's accuracy. This score may be interpreted as a weighted average of the precision and defined as:

$$F1 - Score = \frac{2TP}{(2TP + FN + FP)} \quad (9)$$

- Precision: The proportion of the correctly predicted positive individuals to the total number of actual positive individuals and calculated as:

$$Precision = \frac{TP}{(TP + FP)} \quad (10)$$

It is required to evaluate classification findings using graphical approaches such as the ROC curve and its general ranking, and the region below, i.e., the AUC.

C. Results Analysis

The confusion matrix enables us to quantify the metrics of our categorization study's results. The confusion matrix for the test cases of the proposed CNN framework is illustrated in Fig. 6.

The suggested approach misclassified 44 of the 1029 test images, with adequate and consistent true negative and true positive scores. As a result, the suggested CNN architecture can effectively classify COVID-19 cases.

Furthermore, the CNN classifier performance evaluation is graphically represented in Fig. 7 in terms of loss and accuracy within the validation and training phases. At epoch number 25, the training, as well as validation accuracy, is 96.6% and 95.3%, respectively. Furthermore, the validation and training loss achieved by the proposed system are 0.205 and 0.102, respectively.

The general accuracy, sensitivity, specificity, F1-score, and precision for every issue of the proposed system are

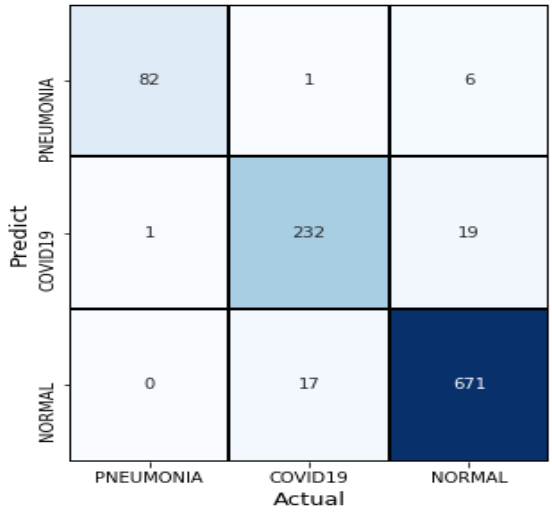


Fig. 6. Confusion Matrix for the Proposed Architecture.

summarized in Table V. The proposed CNN has accuracy 96.3%, sensitivity 93%, specificity 97.4%, F1-score 92%, and precision 92% for COVID-19 cases. For normal cases, accuracy 99.2%, sensitivity 99%, specificity 99.3%, F1-score 92%, and precision 95% have been recorded. The pneumonia cases have obtained an accuracy of 95.9%, sensitivity 96%, specificity 94.9%, F1-score 97%, and precision 98%. The highest accuracy, specificity, and sensitivity are obtained within the normal cases. For example, the best F1-Score and precision have been found in pneumonia cases, along with the highest specificity value.

For better understanding, the classification measurements for every class in terms of accuracy, specificity, precision, recall, and F1-score are shown visually in Fig. 8.

In addition, ROC curves are presented between the false positive rate and thus the true positive rate to check the general performance, which is shown in Fig. 9. The AUC has been determined to be 96.6% for the proposed CNN architecture.

Experimental findings show that the proposed architecture has achieved 96.3% accuracy, 96% AUC, 97.4% specificity, 92% F1-score, 92% precision, and 93% sensitivity for the COVID-19 infected cases.

1) Screening for Pathogens: Grad-CAM is often referred to as a heat map that uses the gradients of an identifying and mapping to view our test dynamically. In order to highlight the significant portions within the image for forecasting, a rough localization map is generated after passing into the last layer. The most important region (processed attributes) from which

TABLE V. PERFORMANCE OF THE PROPOSED NETWORK (IN %)

Class	Accuracy	Sensitivity	Specificity	Precision	F1-Score
Normal	99.2	99	99.3	92	95
Pneumonia	95.9	96	94.9	98	97
Covid19	96.3	93	97.4	92	92

TABLE VI. COMPARATIVE ANALYSIS OF THE PROPOSED ARCHITECTURE WITH EXISTING ONES IN TERMS OF THE SELECTED PERFORMANCE METRICS (IN %)

Author	Accuracy	Specificity	Sensitivity	F1-Score	Precision
Alqudah et al. [14]	95.2	100	93.3	92	90
Kumar et al. [15]	95	96	95	95	95
Apostolopoulos et al. [18]	94.72	96.46	98.66	91	91.5
Khan et al. [20]	95	97.5	96.9	95.6	95
Hemdan et al. [21]	90	89	90	90	83
Sing et al. [22]	96	92	95	96	91
Li et al. [23]	90	93	90	90	91
Proposed System	**96.3**	**97**	**96**	**96**	**96**

TABLE VII. COMPARISON OF THE PROPOSED METHOD WITH EXISTING WORKS REGARDING DETECTION ACCURACY

Authors	Classes	Dataset	Accuracy(%)
Alqudah et al. [14]	2-classes	23 NonCOVID-19/48 COVID-19	95.2%
Apostolopoulos et al. [18]	3-classes	700 Pneumonia/504 Normal/224 COVID-19	93.48%
Hemdan et al. [21]	2-classes	25 non-COVID-19/25 COVID-19	90%
R Kumar et al. [15]	3-classes	1345 Pneumonia/1341 Normal/62 COVID-19	90%
Sethy and Behera [43]	3-classes	127 Pneumonia/127 Normal/127 COVID-19	95.33%
Panwar et al. [44]	2-classes	142 Normal/142 COVID-19	88%
Wang and Wong [11]	3-classes	8066 Pneumonia/5538 Normal/358 COVID-19	93.3%
Ozturk et al. [45]	3-classes	500 Pneumonia/500 Normal/125 COVID-19	87.02%
Proposed System	**3-classes**	**4200 Pneumonia/3218 Normal/2875 COVID-19**	**96.3%**

the classification determination has been made by the network are shown in a deep blue colour. Fig. 10 shows the heat map for normal, COVID-19, and pneumonia cases of classified test samples.

D. Comparative Assessment

The investigation of the outcomes demonstrates that a CNN architecture is highly effective in detecting COVID-19 by sustaining automated feature removal from CXR images. As a result, our proposed approach can separate COVID-19 from normal and pneumonia cases with significant accuracy.

Moreover, to compare the presented framework with existing frameworks, we have re-constructed the methods used in [15], [20], [22] and [23]. All the models are evaluated on the same training, validation, and testing data set to guarantee the fairness of the comparison. Table VI shows a comparative analysis of our system with the above works. Graphical comparisons of the proposed approach with [15], [20], [22] and [23] is shown in Fig. 11.

To assess the performance of our suggested system, we compared the execution results of some existing models to the results of our model, as shown in Table V.

As shown in Table VII, our suggested COVID-19 detection model outperformed a number of other state-of-the-art detection frameworks. Note that it is not feasible to check the performance of the suggested framework with other existing frameworks because of the number of test images; furthermore, because the data sources do not seem to be the same. Accordingly, the correlation between these frameworks is for expository purposes. However, it demonstrates the capability of our methodology in this undertaking. From Table VII, we can state that the results of our proposed framework are generally better than most of the existing frameworks. More specifically, our framework achieves the highest accuracy, F1-score and precision, while it is within a percentage point of the maximum specificity and sensitivity achieved by other methods.

V. CONCLUSION

This study designed a deep CNN-based system to classify chest X-ray images to detect COVID-19 cases, pneumonia, and healthy cases. To identify COVID-19 cases, we used CNN as a motif identifier. The proposed approach has successfully isolated COVID-19 cases from the normal cases and can produce 97% specificity, 96.3% accuracy, 96% sensitivity, 96% F1-score, and 96% precision. The empirical result shows that our proposed framework achieves better accuracy than existing works, and the radiologists assess these performances. In addition, the system is ready to be tested with a more extensive database. It can be employed in local areas where people are affected by the COVID-19 to overcome the lack of expert radiologists.

The suggested framework has a limitation in using a small number of X-ray images for the COVID-19 cases. Future work, which is already under way at the time of writing, includes running our experiments on more such X-ray images from remote hospitals to make our model robust and effective.

ACKNOWLEDGMENT

This work was partially supported by the Italian Ministry of Education and Research (MIUR) in the framework of the CrossLab project (Departments of Excellence).

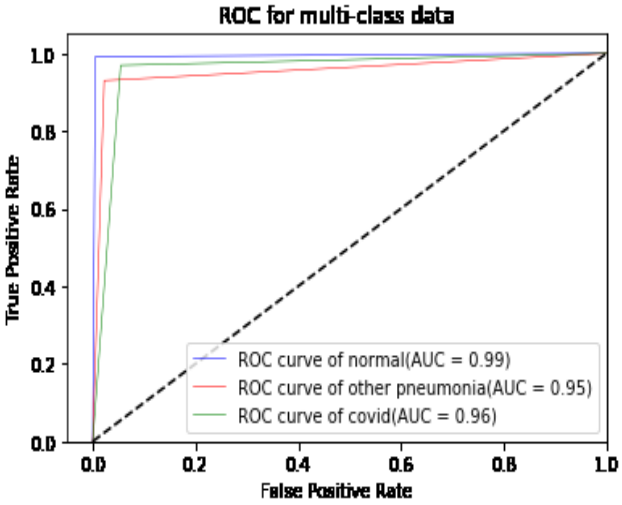


Fig. 9. ROC Analysis of the Proposed Architecture.

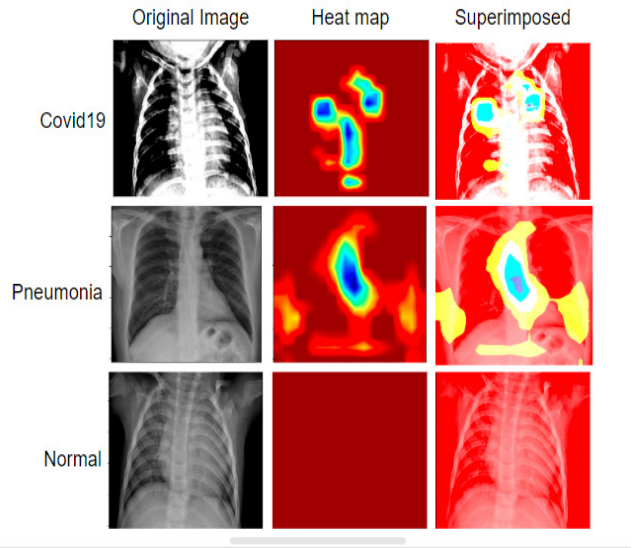
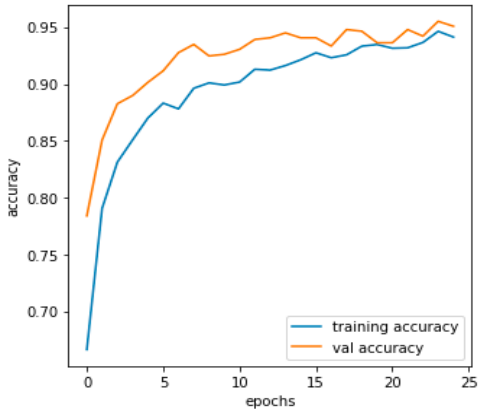
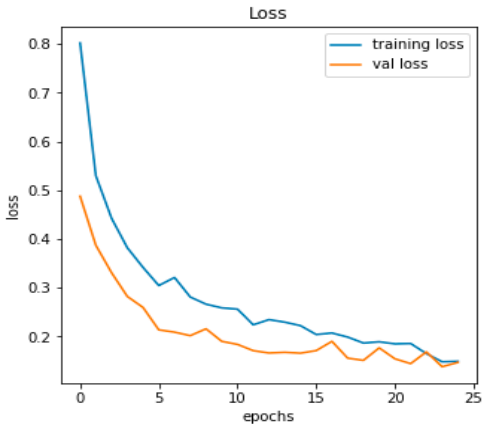


Fig. 10. Heat Map Image Created by the Proposed Architecture.



(a)



(b)

Fig. 7. Performance Evaluation Metrics of the Suggested Approach (a) Accuracy Graph, (b) Loss Graph.

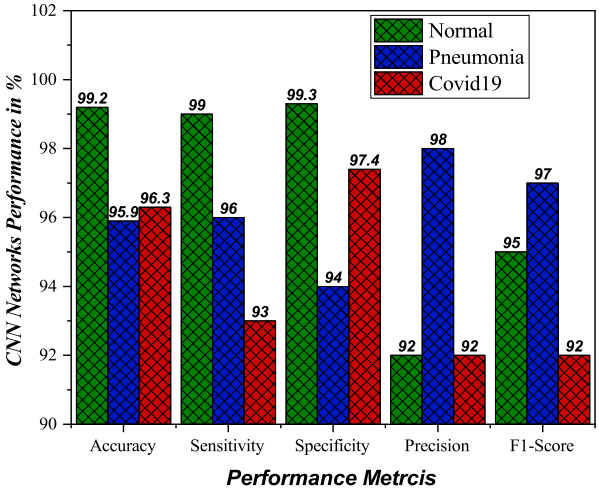


Fig. 8. Performance of the Proposed CNN.

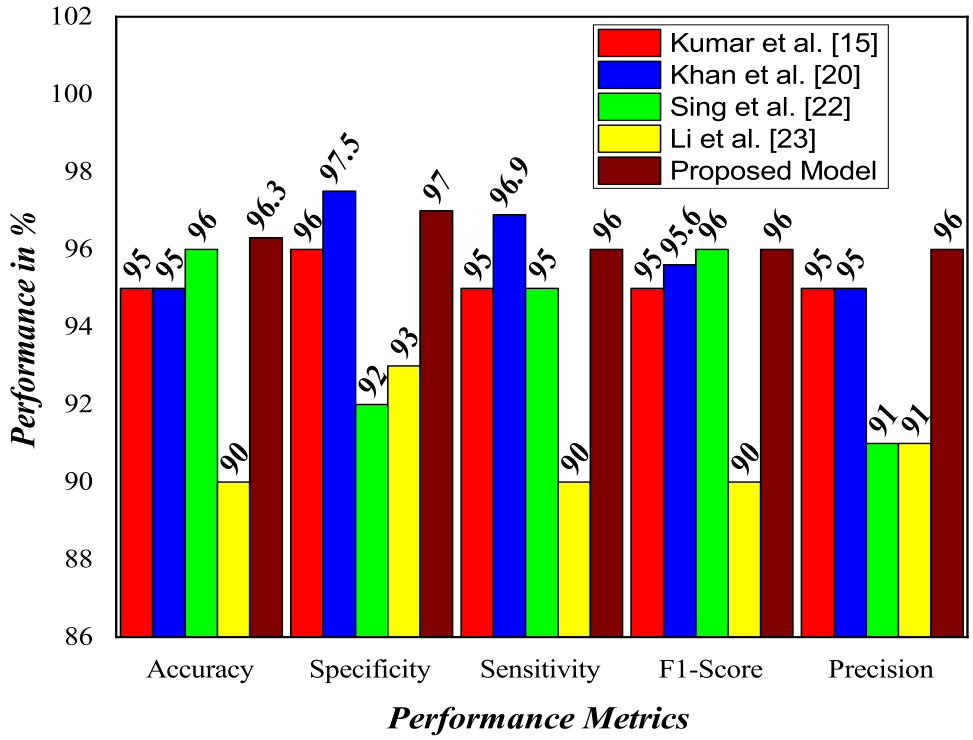


Fig. 11. Graphical Comparison of the Proposed Model with [15], [20] and [23] when Trained and Tested on our Datasets.

REFERENCES

[1] T. Singhal, "A review of coronavirus disease-2019 (covid-19)," *The Indian Journal of Pediatrics*, pp. 1–6, 2020.

[2] C.-C. Lai, T.-P. Shih, W.-C. Ko, H.-J. Tang, and P.-R. Hsueh, "Severe acute respiratory syndrome coronavirus 2 (sars-cov-2) and corona virus disease-2019 (covid-19): the epidemic and the challenges," *International journal of antimicrobial agents*, p. 105924, 2020.

[3] J. Li, J. J. Li, X. Xie, X. Cai, J. Huang, X. Tian, and H. Zhu, "Game consumption and the 2019 novel coronavirus," *The Lancet Infectious Diseases*, vol. 20, no. 3, pp. 275–276, 2020.

[4] J. M. Sharfstein, S. J. Becker, and M. M. Mello, "Diagnostic testing for the novel coronavirus," *Jama*, vol. 323, no. 15, pp. 1437–1438, 2020.

[5] F. A. Rabi, M. S. Al Zoubi, G. A. Kasasbeh, D. M. Salameh, and A. D. Al-Nasser, "Sars-cov-2 and coronavirus disease 2019: what we know so far," *Pathogens*, vol. 9, no. 3, p. 231, 2020.

[6] A. York, "Novel coronavirus takes flight from bats?" *Nature Reviews Microbiology*, vol. 18, no. 4, pp. 191–191, 2020.

[7] M. Loey, F. Smarandache, and N. E. M Khalifa, "Within the lack of chest covid-19 x-ray dataset: A novel detection model based on gan and deep transfer learning," *Symmetry*, vol. 12, no. 4, p. 651, 2020.

[8] V. M. Corman, O. Landt, M. Kaiser, R. Molenkamp, A. Meijer, D. K. Chu, T. Bleicker, S. Brünink, J. Schneider, M. L. Schmidt *et al.*, "Detection of 2019 novel coronavirus (2019-ncov) by real-time rt-pcr," *Eurosurveillance*, vol. 25, no. 3, p. 2000045, 2020.

[9] W.-j. W. Liu, C. Yuan, M.-l. Yu, P. Li, and J.-b. Yan, "Detection of novel coronavirus by rt-pcr in stool specimen from asymptomatic child, china," *Emerg Infect Dis J*, 2020.

[10] W. Wang, Y. Xu, R. Gao, R. Lu, K. Han, G. Wu, and W. Tan, "Detection of sars-cov-2 in different types of clinical specimens," *Jama*, vol. 323, no. 18, pp. 1843–1844, 2020.

[11] L. Wang, Z. Q. Lin, and A. Wong, "Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images," *Scientific Reports*, vol. 10, no. 1, pp. 1–12, 2020.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[13] M. Rahimzadeh and A. Attar, "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," *Informatics in medicine unlocked*, vol. 19, p. 100360, 2020.

[14] A. M. Alqudah, S. Qazan, H. Alquran, I. A. Qasmieh, and A. Alqudah, "Covid-2019 detection using x-ray images and artificial intelligence hybrid systems," *Biomedical Signal and Image Analysis and Project; Biomedical Signal and Image Analysis and Machine Learning Lab: Boca Raton, FL, USA*, 2019.

[15] R. Kumar, R. Arora, V. Bansal, V. J. Sahayasheela, H. Buckchash, J. Imran, N. Narayanan, G. N. Pandian, and B. Raman, "Accurate prediction of covid-19 using chest x-ray images through deep feature learning model with smote and machine learning classifiers," *medRxiv*, 2020.

[16] M. J. Horry, M. Paul, A. Ulhaq, B. Pradhan, M. Saha, N. Shukla *et al.*, "X-ray image based covid-19 detection using pre-trained deep learning models," *engrXiv*, 2020.

[17] F. Ucar and D. Korkmaz, "Covidiagnosis-net: Deep bayes-squeezenet based diagnostic of the coronavirus disease 2019 (covid-19) from x-ray images," *Medical Hypotheses*, p. 109761, 2020.

[18] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine*, p. 1, 2020.

[19] S. K. Bandyopadhyay and S. Dutta, "Machine learning approach for confirmation of covid-19 cases: Positive, negative, death and release," *medRxiv*, 2020.

[20] A. I. Khan, J. L. Shah, and M. M. Bhat, "Coronet: A deep neural network for detection and diagnosis of covid-19 from chest x-ray images," *Computer Methods and Programs in Biomedicine*, p. 105581, 2020.

[21] E. El-Din Hemdan, M. A. Shouman, and M. E. Karar, "Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray

images," *arXiv*, pp. arXiv–2003, 2020.
[22] M. Singh, S. Bansal, S. Ahuja, R. K. Dubey, B. K. Panigrahi, and N. Dey, "Transfer learning based ensemble support vector machine model for automated covid-19 detection using lung computerized tomography scan data," *Research Square*, 2020.
[23] X. Li, C. Li, and D. Zhu, "Covid-mobilexpert: On-device covid-19 screening using snapshots of chest x-ray, 2020," 2020.
[24] S. Ahuja, B. K. Panigrahi, N. Dey, V. Rajinikanth, and T. K. Gandhi, "Deep transfer learning-based automated detection of covid-19 from lung ct scan slices," *Applied Intelligence*, vol. 51, no. 1, pp. 571–585, 2021.
[25] S. J. Fong, G. Li, N. Dey, R. G. Crespo, and E. Herrera-Viedma, "Composite monte carlo decision making under high uncertainty of novel coronavirus epidemic using hybridized deep learning and fuzzy rule induction," *Applied Soft Computing*, p. 106282, 2020.
[26] P. Afshar, S. Heidarian, F. Naderkhani, A. Oikonomou, K. N. Plataniotis, and A. Mohammadi, "Covid-caps: A capsule network-based framework for identification of covid-19 cases from x-ray images," *Pattern Recognition Letters*, vol. 138, pp. 638–643, 2020.
[27] H. Panwar, P. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, and V. Singh, "A deep learning and grad-cam based color visualization approach for fast detection of covid-19 cases using chest x-ray and ct-scan images," *Chaos, Solitons & Fractals*, vol. 140, p. 110190, 2020.
[28] R. Karthik, R. Menaka, and M. Hariharan, "Learning distinctive filters for covid-19 detection from chest x-ray using shuffled residual cnn," *Applied Soft Computing*, vol. 99, p. 106744, 2021.
[29] J. P. Cohen, P. Morrison, and L. Dao, "Covid-19 image data collection," *arXiv 2003.11597*, 2020. [Online]. Available: https://github.com/ieee8023/covid-chest xray-dataset
[30] P. Patel, "Chest x-ray images (pneumonia)," 2020. [Online]. Available: https://www.kaggle.com/prashant268/ches t-xray-covid19-pneumonia
[31] P. Mooney, "Chest x-ray images (pneumonia)," 2017. [Online]. Available: https://www.kaggle.com/paultimothy mooney/chest-xray-pneumonia
[32] F. Pasa, V. Golkov, F. Pfeiffer, D. Cremers, and D. Pfeiffer, "Efficient deep network architectures for fast chest x-ray tuberculosis screening and visualization," *Scientific reports*, vol. 9, no. 1, pp. 1–9, 2019.
[33] D. Singh, V. Kumar, and M. Kaur, "Classification of covid-19 patients from chest ct images using multi-objective differential evolution–based convolutional neural networks," *European Journal of Clinical Microbiology & Infectious Diseases*, pp. 1–11, 2020.
[34] H. Kutlu and E. Avcı, "A novel method for classifying liver and brain tumors using convolutional neural networks, discrete wavelet transform and long short-term memory networks," *Sensors*, vol. 19, no. 9, p. 1992, 2019.
[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.
[36] P. Chang, J. Grinband, B. Weinberg, M. Bardis, M. Khy, G. Cadena, M.-Y. Su, S. Cha, C. Filippi, D. Bota *et al.*, "Deep-learning convolutional neural networks accurately classify genetic mutations in gliomas," *American Journal of Neuroradiology*, vol. 39, no. 7, pp. 1201–1207, 2018.
[37] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.
[38] V. López, A. Fernández, J. G. Moreno-Torres, and F. Herrera, "Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification. open problems on intrinsic data characteristics," *Expert Systems with Applications*, vol. 39, no. 7, pp. 6585–6608, 2012.
[39] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
[40] G. King and L. Zeng, "Logistic regression in rare events data," *Political analysis*, vol. 9, no. 2, pp. 137–163, 2001.
[41] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.
[42] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.
[43] P. K. Sethy and S. K. Behera, "Detection of coronavirus disease (covid-19) based on deep features, 2020," 2020.
[44] H. Panwar, P. Gupta, M. K. Siddiqui, R. Morales-Menendez, and V. Singh, "Application of deep learning for fast detection of covid-19 in x-rays using ncovnet," *Chaos, Solitons & Fractals*, p. 109944, 2020.
[45] T. Ozturk, M. Talo, E. A. Yildirim, U. B. Baloglu, O. Yildirim, and U. R. Acharya, "Automated detection of covid-19 cases using deep neural networks with x-ray images," *Computers in Biology and Medicine*, p. 103792, 2020.