New Generation Computing (2023) 41:533–551
https://doi.org/10.1007/s00354-023-00220-7

Performance Evaluation of Learning Models for the Prognosis of COVID-19

Baijnath Kaushik[1] · Akshma Chadha[1] · Reya Sharma[1]

Received: 5 May 2022 / Accepted: 4 May 2023 / Published online: 24 May 2023

Abstract
COVID-19 has developed as a worldwide pandemic that needs ways to be detected. It is a communicable disease and is spreading widely. Deep learning and transfer learning methods have achieved promising results and performance for the detection of COVID-19. Therefore, a hybrid deep transfer learning technique has been proposed in this study to detect COVID-19 from chest X-ray images. The work done previously contains a very less number of COVID-19 X-ray images. However, the dataset taken in this work is balanced with a total of 28,384 X-ray images, having 14,192 images in the COVID-19 class and 14,192 images in the normal class. Experimental evaluations were conducted using a chest X-ray dataset to test the efficacy of the proposed hybrid technique. The results clearly reveal that the proposed hybrid technique attains better performance in comparison to the existing contemporary transfer learning and deep learning techniques.

Keywords Chest X-ray · COVID-19 · Deep transfer models · Resnet-50 · VGG-16 · VGG-19 · CNN

1 Introduction

COVID-19 is a novel infectious disease caused by viruses. The first case of COVID-19 was found in 2019, and the disease spread rapidly, affecting the lives of people worldwide. One of the most cost-effective alternatives for diagnosing COVID-19 in its early stages is the use computer-aided diagnostic (CAD) techniques, like the X-ray-based procedure for the chest. Coronavirus disease 2019 (COVID-19), typically known as COVID-19, is a communicable illness caused due to coronavirus 2 that results in severe acute respiratory syndrome (SARS-CoV-2) [1]. China's Wuhan City has been the site of the first known case, which

✉ Reya Sharma
sharmareya327@gmail.com

1 School of Computer Science and Engineering, Shri Mata Vaishno Devi University, Katra, India

was identified in December 2019 [2]. There is a continuing worldwide pandemic as a result of the spread of the illness since that time.

Respiratory infection diseases and 2019-nCoV acute respiratory illness were proposed by the WHO in January 2020 as interim names for the virus and disease, following the WHO's 2015 recommendations and international standards against using geographical locations (example Wuhan, China), animal species, or groups of people in disease and viral names, in part to avoid social stigma. These were officially renamed as COVID-19 and SARS-CoV-2 by the WHO on February 11, 2020 [3].

To adequately train deep learning models for specific medical pathologies such as COVID-19, it is necessary to get a sufficiently large publicly available corpus of medical imaging sample data, which is difficult because of the time and resources required to collect and identify images. The "transfer learning" approach of training deep learning models is an alternative to "deep learning networks". Under certain conditions, this approach has been shown to outperform fully trained networks and is commonly used to initialize deep learning models that are then fine-tuned using a sample dataset.

This work aims to design a framework to enable the identification of COVID-19 utilizing image classification and deep learning models for multiple imaging modalities, such as X-ray [4]. This research will examine how transfer learning can be utilized to detect COVID-19 in medical imaging, such as X-ray imaging. This could help health-care providers and researchers design a tool to help them decide on a treatment plan while they are under a lot of pressure.

X-ray imaging is widely used to diagnose pulmonary infections. It is also useful in detecting COVID-19. According to researchers, people with COVID-19 tend to possess patchy infiltrates or opacities that seem similar to other viral pneumonia characteristics. In the early phases of COVID-19, X-ray imaging did not reveal any anomalies. In addition, as the disease progresses, COVID-19 eventually establishes the characteristic unilateral patchy infiltration, affecting the lower, middle, and upper lungs, with occasional signs of consolidation [4].

According to several studies published in the Radiology journal, chest scans may also be useful in diagnosing COVID-19. patients' lungs had visual characteristics such as ground-glass opacification and foggy dark areas that can help doctors tell them apart from people who have not been infected [5, 6]. Findings demonstrate that a chest radiology system can be an efficient technique for the identification, monitoring, and diagnosis of COVID-19.

As COVID-19 continues to spread, many countries are running out of resources. No positive cases must be unreported during this global epidemic. A hybrid classification technique to identify COVID-19 infections in chest radiography images is suggested in this paper, taking the aforementioned into consideration. In this study, the authors implemented VGG-16, VGG-19, CNN and ResNet-50 and proposed a hybrid model (VGG-16 + VGG-19 + CNN) enabling X-ray imaging-based detection of COVID-19. Since radiography images provide a consistent supply of COVID-19, it indicates that deep learning models can help combat the pandemic in the near future.

1.1 Highlights

This study presents deep transfer learning methods for the classification of COVID-19 using chest X-ray datasets. The highlights of this study are summarized as follows.

(a) The dataset taken is balanced with a total of 28,384 X-ray images, having 14,192 images in the COVID-19 class and 14,192 images in the normal class. However, the work done previously contains a very less number of COVID-19 X-ray images.
(b) The authors implemented various deep transfer learning classification models (VGG-16, VGG -19, ResNet-50, and CNN).
(c) A hybrid deep transfer learning-based technique is introduced to detect COVID-19 from a chest X-ray image dataset. The hybrid model is formed by combining three deep learning techniques: CNN, VGG-16, and VGG-19.
(d) In the proposed hybrid model, the authors concatenated the outputs of the flattened layers obtained from VGG-16, VGG-19, and CNN, and two dense layers having 512 and 256 neurons, respectively, were applied to obtain the final classification results.
(e) In this study, the authors focused on evaluating deep transfer learning models and proposed a hybrid model based on hyperparameter tuning.
(f) Experimental evaluations using $k = 5$-fold cross-validation were conducted to test the efficacy of the implemented deep transfer learning and the proposed hybrid model.

The remainder of this paper is organized as follows. Section 2 summarizes the related previous works. Section 3 presents the methodology and approaches employed in this study for classifying X-ray images as COVID-19 and normal. Section 4 describes the experimental strategy and outcomes obtained through the use of deep transfer learning models. The final section discusses the conclusion and provides some future directions.

2 Related Work

Ardakani et al. [7] used artificial intelligence techniques for the purpose of identifying COVID-19 and putting forward valid methods. The author used 108 laboratory-tested data and 1020 CT slabs for COVID-19 patients, and for the non-COVID-19 group 86 patients' data were used. The researcher proposed the various CNN (convolutional neural networks) for discerning the infectious patients from non-COVID-19 patients, such as VGG-19, VGG-16, GoogleNet, SqueezeNet, Xception, AlexNet, MobileNet-V2, ResNet-101 and ResNet-50. Among all the models, ResNet-101 achieved good accuracy. ResNet-101 achieved an area of curve equal to 0.994%. However, the performance was moderated by the radiologist, with an

AUC of 0.873. They suggested that ResNet-101 may be used as a highly susceptible model to characterize and identify COVID-19 infectivity. Since medical imaging is neither used nor recommended for COVID-19 detection in Canada, early spotting of COVID-19 can be done with the help of computer technology and assist in keeping an eye on the progress of the disease, eventually reducing COVID mortalities.

Kassani et al. [8] compared computer frameworks for COVID-19 classification. Neural networks such as MobileNet, DenseNet, Xception, ResNet, InceptionV3, Inception ResNetV2, VGGNet, and NASNet were used to obtain the most accurate features. Several machine learning classifiers using the extracted features classified subjects as either a case of COVID-19 or non-COVID-19 (a control). This technique improves the generalization ability of unseen data and avoids task-specific data pre-processing. The dataset used by the author consists of CT scan images and chest X-rays, and the proposed model performance using neural network was endorsed using the respective dataset. The best performance of the DenseNet-121 feature extractor along with the bagging tree classifier attained an accuracy of 99%. The second-best learner, which is a hybrid of the ResNet50 feature extractor prepared across light GBM, gave an accuracy of 98%.

Heidari et al. [9] used openly available datasets of chest X-ray images where pseudo-color images can be derived from original images and filtered images. The author splits the dataset into three, testing, validation, and training, wherein every class was tested using the convolutional neural network model. The CNN-based CAD strategy yielded a total accuracy of 94.5% and a 95% confidence level in classifying the three classes. The authors demonstrated that improving the accuracy for identification of COVID-19 in chest X-ray images can be done by two image pre-processing steps and the origination of pseudo-color images.

Polsinelli et al. [10] proposed a light CNN design for the proficient bias of COVID-19 CT images to other community-acquired pneumonia and/or CT images of healthy patients. The author's model attains 85.03% accuracy with an upgradation of approximately 3.2% of the initial dataset and approximately 2.1% in the next arrangement of the dataset. They suggested that despite the fact that the gain was low, it is very important in the diagnosis of COVID-19. The author listed the requirement for implementing the CNN model that it can be implemented on a medium-end computer and along with 7.81 GPU acceleration. The researcher recommended that through structured pre-processing approaches, the model performance can be improvised.

COVID-19 has grown all across the world. It emerged in early December of 2019 and has now taken over the whole world, as 5.1 billion people have been affected by this virulent disease till now. Hospitals around the globe were not ready and not furnished in terms of testing kits, to cope with its spreading capability, since physical testing is sluggish and exasperating, i.e., RT-PCR. Due to the manual process, it is necessary to develop early identification and computer-based automated systems for diagnosis purposes. In modern advancements, deep learning algorithms and artificial intelligence have grown in this area together with chest X-ray images.

Nayak et al. [11] use automated techniques of deep learning approaches with X-ray images for the prior stage identification of COVID-19 disease. The number of significant features was compared in search for the best model by the author, such as batch size and learning rate. Validation of these models was done on the basis of publicly accessible data (chest X-ray images). ResNet-34 was identified as the best performer, with an accuracy of 98.33%.

Farooq and Hafeez [12] presented a precise framework of CNN to differentiate other pneumonia cases from contagious COVID-19 cases by using publicly available datasets. They worked on a three-step method to fine-tune a pre-trained ResNet-50 model to enhance the overall performance of the model and decrease training time, which they named COVID-ResNet, wherein progressively re-sizing of input images to $128\times128\times3$, $224\times224\times3$, and $229\times229\times3$ pixels and at each stage fine-tuning were executed. By automatic learning rate selection, along with 41 epochs, the achieved performance accuracy was 96.23%. The author has developed the model in terms of discerning the normal non-infectious individuals with three distinct infection types.

Sitaula and Hossain [13] proposed an attention-based deep learning model with an attention module through VGG-16. The regions of interest in X-ray images of the chest are a kind of spatial relationship is established amid the regions by adopting the attention module. Further, adding the apt convolution layer (4th pooling layer) from the VGG-16, the deep learning model designed by attention module in order to apply fine-tuning for classification process. In this work, the authors have used three datasets in order to evaluate the performance of the model. The proposed methodology gives favorable classification results in comparison to other methods applicable for CXR image dataset of COVID-19.

A lightweight CNN model named LW-CORONet is proposed by Soumya Ranjan Nayak et al. It consists of two fully connected layers after each convolution, rectified linear unit (ReLU), and pooling layer. Using just five learnable layers, the proposed model makes it easier to extract significant features from chest X-ray (CXR) images. We assess the suggested model using two larger CXR datasets. The effects of several hyperparameters, including batch size, learning rate, and various optimization methods, were investigated [14].

To identify COVID-19 patients using real-world datasets, Moutaz Alazab et al. analyzed the incidence of COVID-19 distribution over the globe and provided a deep convolutional neural network (CNN)-based artificial intelligence technique. The suggested system's utility in detecting COVID-19 was proven by empirical results acquired from 1000 X-ray pictures of actual patients. In addition, three forecasting techniques were used to forecast the number of COVID-19 confirmations, recoveries, and deaths over the following 7 days: the prophet algorithm (PA), autoregressive integrated moving average (ARIMA) model, and long short-term memory neural network (LSTM). According to the projection findings, Australia and Jordan both performed well [15] (Table 1).

3 Methodology

3.1 Dataset

In this study, the COVID dataset [18] comprises chest X-ray images. The chest X-ray dataset was categorized into two classes: normal and COVID-19. The chest

Table 1 Summary and limitations of related work done in the past

Authors, year	Deep learning method used	Contribution	Limitations
Ardakani et al., 2020 [7]	VGG-16, AlexNet, SqueezeNet, VGG-19, MobileNet-V2, GoogleNet, etc., were used	An accuracy of 99.51% is obtained by ResNet-101, while using Xception the accuracy achieved is 99.02%	The explanation of the dataset is missing the study. The models used are evaluated on a single dataset which doubts the significance of the respective models
Kassaniet al., 2021 [8]	Models such as InceptionResNetV2, VGG-Net, NASNet, etc., were used	DenseNet classifier achieved the best accuracy of 99% while using it with bagging classifier	The results obtained before and after hyper-parameter tuning are not explained
Heidariet al. 2021 [9]	CNN-based CAD	The computer-aided diagnosis (CAD) scheme shows a total of 94.5% accuracy	The use of larger datasets should be incorporated for performance evaluation of the proposed model
Polsinelli et al. 2020 [10]	Light CNN	An accuracy of 85.03% with an improvement of about 3.2% is obtained	Efficient pre-processing techniques should be used
Nayak et al. 2021 [11]	VGG-16, SqueezeNet, GoogleNet, AlexNet, ResNet-34, MobileNet-V2, etc.	The optimal performance was achieved by ResNet-34, which showed 98.33% accuracy	The efficiency of the model can be confirmed if the multiclass classification problems is also incorporated
Farooq and Hafeez, 2021 [12]	ResNet-50	An accuracy of 96.23% is obtained	The model proposed needs to be evaluated on other related larger datasets as well to ensure the accuracy of the model
Sitaula and Hossain, 2021 [13]	VGG-16	The proposed VGG-16 model gives an accuracy of 79.58%	The performance can be improved using an autoencoder before training the model
Keleset al., 2021 [16]	CNNet and ResNet	Accuracies of 97.61% and 94.28% are shown for CNNet and ResNet, respectively	The consistency of the proposed model can be improved by employing it on other such datasets as well
Tan et al., 2021 [17]	VGG-16	Accuracies of 79.58, 85.43% and 87.49 are obtained on three kind of datasets	Due to overfitting problem, the accuracies obtained are higher

Table 2 Detailed sample-wise distribution of the dataset

Dataset	Normal	COVID-19
Chest X-ray	14,192	14,192

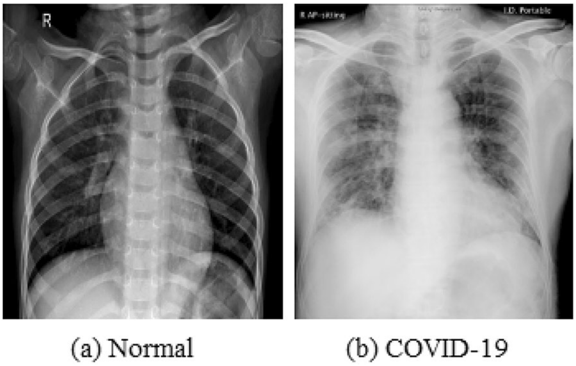

Fig. 1 Sample images from the COVID-19 chest X-ray database [19]

X-ray dataset contains a total of 28,384 posterior chest radiography images, and the detailed distribution of the dataset in terms of the number of samples in each category is shown in Table 2.

The publicly available dataset repositories were used to collect the dataset [19, 20]. The dataset taken is balanced with a total of 28,384 X-ray images having 14,192 images in the COVID-19 class and 14,192 images in the normal class. However, the work done previously contains a very less number of COVID-19 X-ray images. The dataset is then divided into training and testing with 23,384 and 5000 images, respectively (Fig. 1).

3.2 Deep Transfer Learning Models

3.2.1 VGG-16

The VGG-16 neural network [21] is one of the first architectures to demonstrate that deeper neural networks with smaller convolutional filters can obtain better results than shallow networks. The remarkable feature of VGG-16 is that instead of having many hyperparameters, it has a much simpler network architecture. The model consists of convolutional layers having 3×3 filters with a stride of 1, and with the same padding. In all the max-pooling layers, 2×2 filters were used with a stride of 2. The first two layers have 64 convolutional filters with a size of 3×3. Therefore, it ends up with a $224\times224\times64$ volume, since we have used the same convolutions (i.e., width and height are the same). After this, pooling is used to reduce the height and width; that is, after applying the pooling layer, the volume increases from $224\times224\times64$ to $112\times112\times64$. Subsequently, a couple of convolutional layers were added to the network. Here, 128 filters were used with the same convolutions, so the new dimension was $112\times112\times128$. Then, a pooling layer was added, which reduced the volume of the network to $56 \times 56 \times 128$. A few more convolutional layers with 256 and 512 filters were added, and after applying the pooling layer, the final volume was $7 \times 7 \times 512$. Finally, a fully connected layer was used with 4096 units.

The number 16 in the name VGG-16 signifies that it has 16 layers with few weights. VGG-16 has a consistent architecture wherein there are some convolutional layers succeeding a pooling layer that decreases the width and height of the volume. It is clearly visible that initially the model uses 64 filters, and then we double it to 128 and then to 256 filters, and 512 filters are used in the last layers. The number of filters were doubled across each stack of convolutional layers, and this basic principle was used to strategize the network architecture. The key disadvantage of VGG-16 is that it has a huge network with a significant number of training parameters.

3.2.2 VGG-19

The VGG-19 network [25] is a pre-trained model proposed by Simonyan and Zisserman in 2014. The VGG-19 model has been reported to achieve significant results in the past in terms of recognition accuracy on image processing datasets such as ImageNet. This model comprises approximately 143 million parameters that are learned from the ImageNet dataset, which consists of 1.2 million samples images belonging to 1,000 different categories of general objects [22]. There are two dense layers, each with 512, 256 neurons and an activation function of 'ReLU'. A dropout function (ratio = 0.5) was used to remove unnecessary neurons from the dense layers. Model overfitting is avoided by employing the 'Adam' optimizer with a 0.001 learning rate and a categorical cross-entropy loss function for many classes. The VGG-19 model comprises 19 trainable layers with convolutional layers, max-pooling layers, fully connected layers, and dropout layers. In the present work, the VGG-19 model was trained on an X-ray image dataset.

3.2.3 ResNet-50

ResNet, which is short for residual networks, is a classic neural network that is used as a backbone for several computer vision tasks. The ResNet model permits us to train extremely deep neural networks with 50 layers successfully. Resnet-50 [23] is a transfer learning model. It is developed to avoid the vanishing gradient problem in deep neural networks by implementing a mode of skip connection between layers. This model architecture results in a network that is more efficient for training and provides good performance in terms of accuracy. In this study, an X-ray image classification dataset with three different classes (COVID19, and normal) was trained using the ResNet-50 model. The ResNet-50 model was pre-trained on the ImageNet dataset. However, in this work, we trained the model on an X-ray image dataset, and the weights were updated accordingly. The fully connected layers of the ResNet-50 model were removed, and new layers were added based on the chosen dataset. The ResNet-50 model has 1000 classes; however, in this study, only 3 classes were used. Therefore, the output layers are updated accordingly. There are four dense layers with 1024, 512, 256, 128 number of neurons, and a rectified linear unit 'ReLU' [24] activation function was used. The unwanted neurons were removed from the dense layers using a dropout function (ratio = 0.5) [25]. To avoid overfitting the model, the Adam optimizer [26] was used with a learning rate of 0.0001. A categorical cross-entropy loss function was used for multiple classes.

3.2.4 Convolutional Neural Network

A convolutional neural network (CNN) was initially designed in 1998 by LeCun et al. [27]. The primary aim of constructing a CNN is to combine feature learning and classification processes. CNNs have recently achieved significant success in the field of pattern recognition and image classification. In this study, a two-dimensional (2D) CNN was used to perform classification tasks on a chest X-ray image dataset. Every input is passed through the 2D convolutional layers, followed by max-pooling layers and dense layers. The proposed CNN model has ten convolutional layers with ten kernels, each kernel size is (3, 3), activation function is 'ReLU', and the padding is the same. Batch normalization is used to remove the overfitting error by converting big data into small batches to train, and the pooling layer size is (2, 2) with strides (2, 2). There are two dense layers, with 512 and 256 neurons, and the 'ReLU' activation function. The sigmoid activation was used in the last dense layer. A dropout rate [25] of 0.02 is used in the current work to avoid the chances of overfitting. The categorical cross-entropy loss and Adam optimizer [26] with a learning rate of 0.001 were used to train the model. Precision and recall functions, as well as the ROC curves, were used to generalize the accuracy of the model. The detailed procedure of the CNN is presented in Algorithm 1.

Algorithm 1: Procedure Convolutional Neural Network (CNN)

1: **Input:** Training dataset (d_{train}), test dataset (d_{test}), training epochs (e), input shape (inp), and batch size (b).
2: **Output:** The output is classified in three categories: Normal, and COVID-19 and the model will return the following metrics: $accuracy, precision, recall, f1\ score$.
3: $image\_Datagenerator \leftarrow Rescale(d_{train}, d_{test})$ *//Perform the data augmentation*
4: $model\_cnn \leftarrow Sequential()$
5: $model\_cnn.add\_Conv2D(kernels, kernel\_size, activation, strides, kernel\_initializer)$
6: $model\_cnn.add(BatchNormalization())$
7: $model\_cnn.add\_MaxPooling2D(pool\_size, stride)$
8: $model\_cnn.add(Dropout(rate))$
9: $model\_cnn.add(Flatten())$
10: $model\_cnn.add(Dense(layers, activation))$
11: $model\_cnn.summary()$
12: $i \leftarrow 1$
13: **while** $i \leq e$ **do**
14: $model\_cnn\_Fit(d_{train}, b, e) \leftarrow$ Train the Convolutional Neural Network Model
15: **end while**
16: $accuracy, precision, recall, f1\ score \leftarrow model\_cnn\_Evaluate(d_{test})$
17: return $accuracy, precision, recall, f1score$

3.3 Hybrid Model

The proposed hybrid model is a combination of VGG-16, VGG-19, and CNN. The outputs obtained from the three best-performing deep learning models (i.e., VGG-16, VGG-19, and CNN) were concatenated to produce the final results. The overall framework of the hybrid approach is presented in Algorithm 2.

Algorithm 2: Procedure Hybrid Model

Input: Training dataset(d_{train}), test dataset(d_{test}), training epochs (e), input shape (inp), and batch size (b).
Output: The output is classified in three categories: Normal, and COVID-19 and the model will return the following metrics: $accuracy, precision, recall, f1\ score$
$image\_Datagenerator \leftarrow Rescale(d_{train}, d_{test})$ *//Perform the data augmentation*
$model\_vgg16 \leftarrow VGG16(input\_shape, weights)$
$model\_vgg19 \leftarrow VGG19(input\_shape, weights)$
$model\_cnn \leftarrow Sequential()$
$vgg16\_x \leftarrow Flatten()(model\_vgg16(inp))$
$vgg19\_x \leftarrow Flatten()(model\_vgg19(inp))$
$cnn\_x \leftarrow Flatten()(model\_cnn(inp))$
$model\_hybrid1 \leftarrow Concatenate(vgg16\_x, vgg19\_x, cnn\_x)$
$i \leftarrow 1$
while $i \leq e$ **do**
$model\_hybrid1\_Fit(d_{train}, b, e) \leftarrow$ Train the Hybrid Model-1
end while
$accuracy, precision, recall, f1\ score \leftarrow model\_hybrid1\_Evaluate(d_{test})$
return $accuracy, precision, recall, f1\ score$

The architecture used for the proposed hybrid approach is shown in Fig. 2. The X-ray image categorization dataset has two distinct classifications (COVID-19, and normal). This dataset is trained using the proposed hybrid model by identifying the patient's disease with the help of X-ray images. In this model, two dense layers are

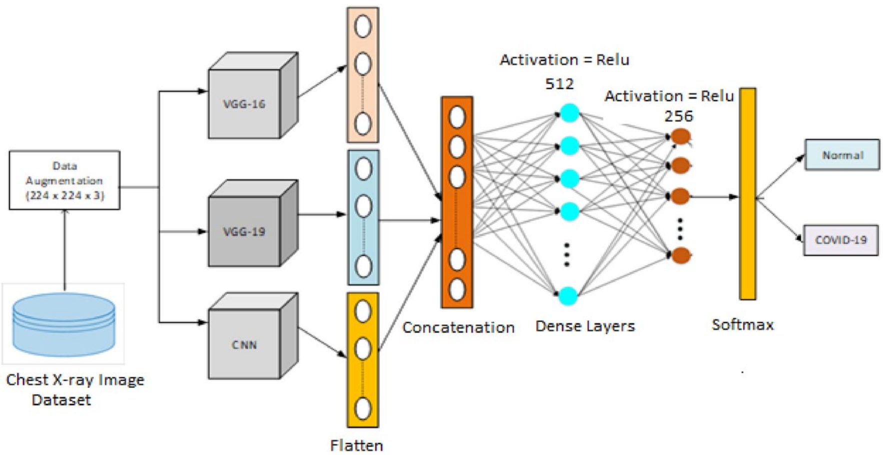

Fig. 2 Architecture of the proposed hybrid approach

used, each with 512 and 256 neurons, and ReLU activation function. A dropout function (ratio = 0.5) is used to remove unnecessary neurons from the dense layers. To avoid model overfitting, the 'Adam' optimizer was employed with a learning rate of 0.001, and a categorical cross-entropy loss function was used while training the model. Precision and recall functions were used to generalize the model's accuracy, as described in Sect. 4.

4 Experimental Analysis

This section presents several experimental analyses of the proposed architectures to evaluate their performance. The authors examined various performance criteria to evaluate the classification problem predictions.

4.1 Performance Metrics

The performance metrics for classifications problems that are most frequently used are as follows:

Accuracy
Confusion matrix
Precision
Recall
F1_Score

4.1.1 Accuracy

Accuracy is a fundamental criterion for classification Accuracy is the ratio of correctly classified points to the total number of points. Accuracy is defined as the percentage of true results compared to the total number of cases evaluated.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}. \tag{1}$$

4.1.2 Precision

Precision is the percentage of correctly classified cases out of the total number of cases classified. The number of correct items retrieved by our ML model can be used as a measure of precision. Using the following formula, we can effectively calculate it using the confusion matrix:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{2}$$

4.1.3 F1_Score

F1_Score determines the harmonic mean of the metrics, recall and precision.

$$F1_{\text{Score}} = \frac{2 \times (\text{Recall} \times \text{Precision})}{\text{Recall} + \text{Precision}}. \tag{3}$$

4.1.4 Recall

This may be defined as the number of positive values returned. We can easily calculate it using a confusion matrix with the help of the following formula:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{4}$$

4.1.5 Matrix of Confusion

The confusion matrix is a summary of expected outcomes in a specific table pattern that enables visualization of the machine learning model's performance metric for a binary classification problem with two classes or a multiclass classification problem with more than two classes.

4.2 Results

The multiclass classification results of the implemented models were recorded for various performance parameters, as follows.

The metrics mentioned above are usually implied to evaluate the efficiency of classification models. The accuracy, precision, recall, and F1-Score for the implemented model are presented in Table 3. In this work, four deep transfer learning models: VGG-16, VGG-19, CNN, and Resnet-50 have been implemented. VGG-19 outperformed other deep transfer learning models with an accuracy of 78.54%, VGG-16 attained an accuracy of 74.44%, CNN achieved an accuracy of 72.72%, and Resnet-50 showed the lowest accuracy of 69.92%.

Table 3 Performance analysis of different deep learning models and the proposed hybrid model on the chest X-ray dataset

Model	Accuracy (%)	Precision	Recall	F1_Score
CNN	72.72	66.20	76.13	70.82
VGG-16	74.44	67.84	78.16	72.64
VGG-19	78.54	70.52	83.99	76.67
Resnet-50	69.92	65.84	71.69	68.64
Proposed hybrid model (VGG-16 + VGG-19 + CNN)	81.42	74.88	86.15	80.12

Further, the three best-performing models (VGG-16, VGG-19, and CNN) were combined and the proposed hybrid model showed the highest accuracy of 81.42% on a chest X-ray imaging database. The other performance parameters for these implemented models are elucidated in Table 3 to analyze how well these models would distinguish between COVID-19 and normal class.

In this study, the authors demonstrated that for given imaging data, the COVID-19 detection tools are constructed and implemented using deep transfer learning models. The proposed hybrid model (VGG-16 + VGG-19 + CNN) is capable of categorizing COVID-19 and normal situations using imaging modalities.

The performance of the proposed model is shown as a confusion matrix (CM) in Fig. 3. True positive (TP) result is interpreted as the model predicting a positive class, which is true. False positive (FP) result is interpreted as the model predicting a positive class, but which is incorrect. False negative (FN) result is interpreted as a model predicting a negative class, but which is inaccurate. True negative (TN) result is true because the model predicts a negative class. Furthermore, a comparison of deep transfer learning models and the proposed hybrid model based on accuracy, precision, and recall is shown in Fig. 4.

TN = 1981	FP = 845
FN = 519	TP = 1655

(a): CNN

TN = 2026	FP = 804
FN = 474	TP = 1696

(b): VGG-16

TN = 2164	FP = 737
FN = 336	TP = 1763

(c): VGG-19

TN = 1850	FP = 854
FN = 650	TP = 1646

(d): ResNet-50

TN = 2199	FP =628
FN = 301	TP = 1872

(e): Hybrid Model

Fig. 3 Confusion matrices of CNN, VGG-16, VGG-19, Restnet-50, and the proposed hybrid model

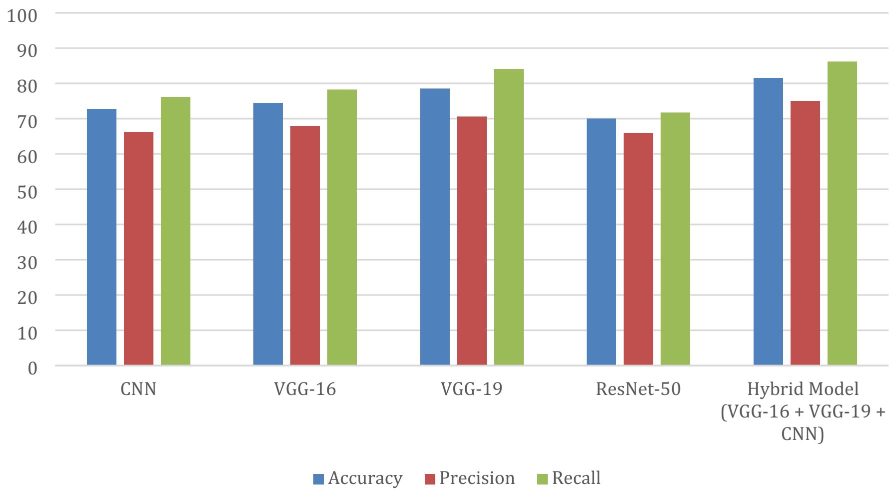

Fig. 4 Comparison among CNN, VGG-16, VGG-19, Restnet-50, and the proposed hybrid model (VGG-16, VGG-19, CNN) based on accuracy, precision, and recall

4.3 Hyperparameter Tuning

A mathematical framework with a set of parameters that are learned from data is known as a machine learning technique. We were able to fit the parameters in the model by training the model with existing data. Furthermore, there is another type of parameter known as hyperparameters that cannot be learned directly from the training procedure. Before the actual training process begins, they are usually rectified. These parameters describe crucial aspects of the model, such as its complexity and learning speed. Hyperparameter tuning was carried out using a grid search in this study, and the optimized parameters are listed in Table 4.

A mathematical model with a number of parameters that must be investigated in the data is characterized as a type of machine learning. We can fit the model's parameters by training the model and using current data. Hyperparameters, on the other hand, are a sort of parameters that cannot be learned directly from a conventional training session. They are usually resolved prior to the start of the training program. These features reflect the model's key qualities, such as complexity and learning speed. Hyperparameters are optimized using grid search in this study, and the best results are given in Table 4. The most extensively used method by data scientists is K-fold cross-validation. This is a data partitioning feature, so the dataset can be efficiently used to build a common model. The main goal of any kind of machine learning is to create a global model that can work well with invisible data. In this study, we use a fivefold cross-validation procedure. The accuracy of cross-validation is shown in Table 5.

The results obtained after applying hyperparameter tuning using grid search on different deep transfer learning models and the proposed hybrid model are

Table 4 Optimal hyperparameters acquired using grid search

Model	Parameters	Best parameters
CNN	1. optimizer = [SGD, RMSprop and ADAM] 2. Loss function = mean squared Error and categorical_crossentropy 3. Batch size = 16, 32 and 64	1. optimizer = [RMSprop] 2. Loss function = categorical_crossentropy 3. Batch size = 16
VGG-16	1. Optimizer = [SGD, RMSprop and ADAM] 2. Loss function = mean squared Error and categorical_crossentropy 3. Batch size = 16, 32 and 64	1. Optimizer = [ADAM] 2. Loss function = categorical_crossentropy 3. Batch size = 32
VGG-19	1. Optimizer = [SGD, RMSprop and ADAM] 2. Loss function = mean squared Error and categorical_crossentropy 3. Batch size = 16, 32 and 64	1. Optimizer = [ADAM] 2. Loss function = categorical_crossentropy 3. Batch size = 16
ResNet-50	1. Optimizer = [SGD, RMSprop and ADAM] 2. Loss function = mean squared Error and categorical_crossentropy 3. Batch size = 16, 32 and 64	1. Optimizer = [RMSprop] 2. Loss function = categorical_crossentropy 3. Batch size = 16
Proposed hybrid model	1. Optimizer = [SGD, RMSprop and ADAM] 2. Loss function = mean squared Error and categorical_crossentropy 3. Batch size = 16, 32 and 64	1. Optimizer = [ADAM] 2. Loss function = categorical_crossentropy 3. Batch size = 16

Table 5 Performance analysis based on the accuracy of cross-validation

Model	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5
CNN	72.42	72.69	73.14	73.51	73.83
VGG-16	74.22	74.76	74.99	75.26	75.45
VGG-19	78.12	78.86	79.51	79.96	80.02
ResNet-50	69.54	69.91	70.52	71.00	71.15
Hybrid	81.17	81.63	82.35	82.72	82.94

shown in Table 6. The performance of the proposed model is also shown in the form of a confusion matrix (CM) in Fig. 5.

Furthermore, a comparison of deep transfer learning models (CNN, VGG-16, VGG-19) and the proposed hybrid model based on accuracy, precision, and recall after hyperparameter tuning is shown in Fig. 6. It can also be observed from the results shown in Fig. 6 that the proposed hybrid model achieves the best performance among all the deep transfer learning models for the detection of COVID-19 from the chest X-ray image dataset.

Table 6 Performance analysis of different deep learning models and the proposed hybrid model on the chest X-ray dataset after applying hyperparameter tuning

Model	Accuracy	Precision	Recall	F1_Score
CNN	73.22	66.72	76.69	71.36
VGG-16	75.30	68.52	79.26	73.50
VGG-19	79.64	71.36	85.52	77.80
ResNet-50	70.92	66.40	73.00	69.54
Proposed hybrid model (VGG-16 + VGG-19 + CNN)	82.48	76.04	87.28	81.27

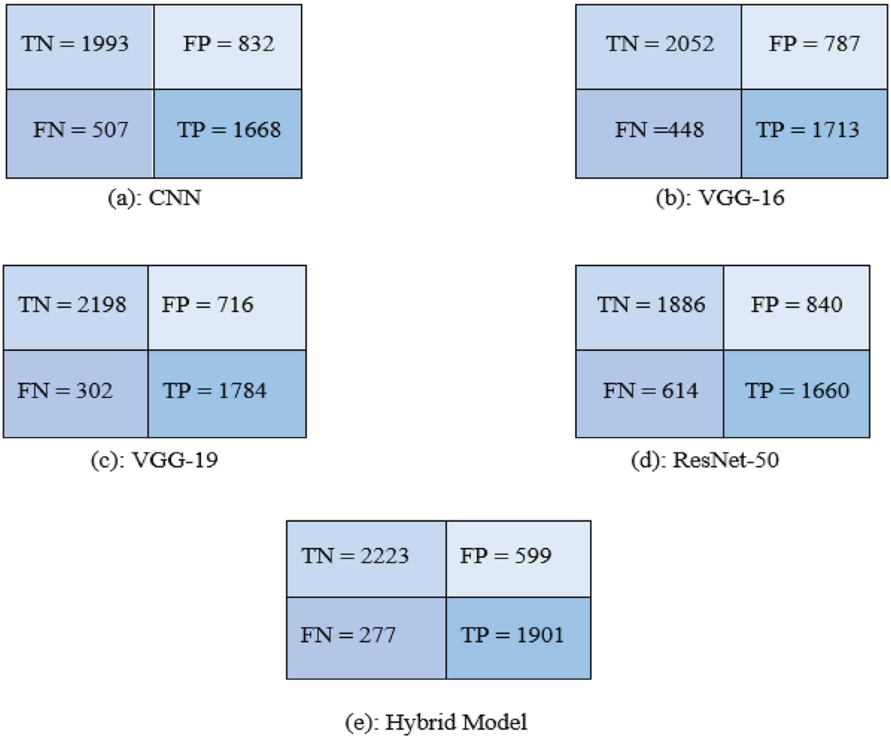

Fig. 5 Confusion matrices of CNN, VGG-16, VGG-19, Restnet-50, and the proposed hybrid model after applying hyperparameter tuning

5 Conclusion

COVID-19 virus has aroused a pandemic that has affected many people and even taken the lives of many. To prevent COVID-19, it must be detected first, to end this pandemic as it is spreading at a very high speed. The COVID dataset used in this study comprised chest X-ray images. The chest X-ray dataset was categorized into two classes: normal and COVID-19. The chest X-ray dataset contained

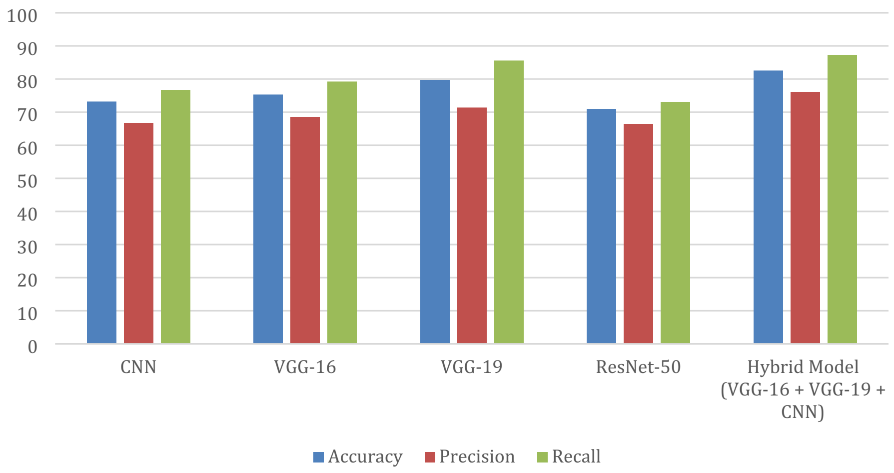

Fig. 6 Performance comparison among CNN, VGG-16, VGG-19, Restnet-50, and the proposed hybrid model (VGG-16, VGG-19, CNN) based on accuracy, precision, and recall after applying hyperparameter tuning

28,384 posterior chest radiography images. In this study, the authors implemented CNN, VGG-16, VGG-19, and ResNet-50 models, and a hybrid deep transfer learning-based model is proposed for the detection of COVID-19 from a chest X-ray image dataset. The hybrid model is formed by combining the three top-performing deep transfer learning models, viz., VGG-16, VGG-19, and convolutional neural network (CNN). Investigational assessments were conducted using a chest X-ray dataset to test the effectiveness of the proposed hybrid model. The results show that the proposed hybrid model achieves better performance and provides better results than existing contemporary deep transfer learning models such as ResNet-50, VGG-16, VGG-19, and CNN with an accuracy of 81.42%. Hyperparameter tuning was performed using grid search, and the optimized parameters were used to again train the model. Fivefold cross-validation was also performed and the results again showed that the hybrid model performed better than all the other models with an accuracy of 82.48%. In future, the authors will perform the work on larger datasets and with more than one dataset to increase the significance of the proposed model. Further, more efficient pre-processing techniques will be employed to increase the efficacy of the model and autoencoders will also be used to train the model. Additionally, we will definitely look forward to further improving the model development for reducing the risk of bias. For example, it would be interesting to explore the possibility of designing a private dataset using either antibody tests or positive RT-PCR and analyze the performance of the proposed model on that dataset.

Data Availability Data will be provided on request to the corresponding author.

Declarations

Conflict of interest The authors declare that they have no conflicts of interest to declare regarding this study.

References

1. Mahase, E.: Coronavirus: COVID-19 has killed more people than SARS and MERS combined, despite lower case fatality rate. (2020)
2. Wang, D., et al.: Clinical characteristics of 138 hospitalized patients with 2019 novel coronavirus-infected pneumonia in Wuhan, China. JAMA **323**(11), 1061–1069 (2020)
3. Wang, W., et al.: Detection of SARS-CoV-2 in different types of clinical specimens. JAMA **323**(18), 1843–1844 (2020)
4. Wang, L., Zhong, Q.L., Wong, A.: Covid-net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest x-ray images. Sci. Rep. **10**(1), 1–12 (2020)
5. Fang, Y., et al.: Sensitivity of chest CT for COVID-19: comparison to RT-PCR. Radiology **296**(2), E115–E117 (2020)
6. Xie, X., et al.: Chest CT for typical coronavirus disease 2019 (COVID-19) pneumonia: relationship to negative RT-PCR testing. Radiology **296**(2), E41–E45 (2020)
7. Ardakani, A.A., Kanafi, A.R., Rajendra Acharya, U., Khadem, N., Mohammadi, A.: Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: results of 10 convolutional neural networks. Comput. Biol. Med. **121**, 103795 (2020)
8. Kassani, S.H., Kassani, P.H., Wesolowski, M.J., Schneider, K.A., Deters, R.: Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: a machine learning based approach. Diabetes Res. Clin. Pract. **41**, 867–879 (2021)
9. Heidari, M., Mirniaharikandehei, S., Khuzani, A.Z., Danala, G., Qiu, Y., Zheng, B.: 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)
10. Polsinelli, M., Cinque, L., Placidi, G.: A light CNN for detecting COVID-19 from CT scans of the chest. Pattern Recogn. Lett. **140**, 95–100 (2020)
11. Nayak, S.R., Nayak, D.R., Sinha, U., Arora, V., Bilas, R., Pachori: Application of deep learning techniques for detection of COVID-19 cases using chest X-ray images: a comprehensive study. Biomed. Signal Process. Control **64**, 102365 (2021)
12. Farooq, M., Hafeez, A.: COVID-ResNet: a deep learning framework for screening of COVID19 from radiographs. (2021)
13. Sitaula, C., Hossain, M.B.: Attention-based VGG-16 model for COVID-19 chest X-ray image classification. Appl. Intell. **51**, 2850–2863 (2021)
14. Nayak, S.R., Nayak, D.R., Sinha, U., Arora, V., Pachori, R.B.: An efficient deep learning method for detection of COVID-19 infection using chest X-ray images. Diagnostics **13**(1), 131 (2023)
15. Alazab, M., Awajan, A., Mesleh, A., Abraham, A., Jatana, V., Alhyari, S.: COVID-19 prediction and detection using deep learning. Int. J. Comput. Inf. Syst. Ind. Manag. Appl. **12**(June), 168–181 (2020)
16. Keles, A., Keles, M. B., Keles, A.: COV19-CNNet and COV19-ResNet: diagnostic inference engines for early detection of COVID-19. Cogn. Comput. (2021)
17. Tan, W., Liu, P., Li, X., Liu, Y., Zhou, Q., Chen, C., Gong, Z., Yin, X., Zhang, Y.: Classification of COVID-19 pneumonia from chest CT images based on reconstructed super-resolution images and VGG neural network. Health Inf. Sci. Syst. **9**, 10 (2021)
18. Wang, L., Wong, A.: Covid-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest X-ray images. (2020).
19. Kaggle link for the dataset—https://www.kaggle.com/datasets/andyczhao/covidx-cxr2?select=competition_test. Accessed 20 Dec 2021
20. GitHub repo of the dataset—https://github.com/lindawangg/COVID-Net/blob/master/docs/COVIDx.md. Accessed 20 Dec 2021
21. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint* arXiv:1409.1556 *(2014).*

22. Zhang, X., Zou, J., He, K., Sun, J.: Accelerating very deep convolutional networks for classification and detection. IEEE Trans. Pattern Anal. Mach. Intell. **38**, 1943–1955 (2016)
23. He, K. et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
24. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning, pp. 807–814 (2010)
25. Srivastava, N., et al.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
26. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. *arXiv preprint.* arXiv:1412.6980 *(2014).*
27. LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time series. Handb. Brain Theory Neural Netw. **3361**(10), 1995 (1995)