International Journal of Computational Intelligence Systems (2024) 17:240
https://doi.org/10.1007/s44196-024-00609-2

RESEARCH ARTICLE

COVID-19 Detection Systems Based on Speech and Image Data Using Deep Learning Algorithms

Farooq Akhtar[1] · Rabbia Mahum[1] · Adham E. Ragab[2] · Faisal Shafique Butt[3] · Mohammed A. El-Meligy[4,5] · Haseeb Hassan[6]

Received: 7 September 2023 / Accepted: 18 July 2024

Abstract
COVID-19 is a worldwide epidemic that seriously affected the lives of people. Since its inception, physicians have tried their best to trace the virus and reduce its spread. Several diagnostic approaches have been reported to detect the coronavirus in research, clinical, and public health laboratories. Although the existing systems aid medical experts in the diagnosis, they still lack precise detection and may fail to detect COVID-19 in a timely manner. Therefore, in this study, we recommend two approaches i.e., the first approach is based on the VGGish network that focuses on vocal signals, such as breathing and coughing, and the second approach is based on ResNet50, which takes chest X-rays as input. With the help of VGGish, the patient's cough, voice, and respiration audios have been classified as patient and non-patient achieving an accuracy of more than 98%. We also assessed the performance of several methods for X-ray classification, such as ResNet50, VGG16, VGG19, Densnet201, Inceptionv3, Darknet, GoogleNet, squeezeNet, and Alex-Net. TheResNet50 outpaced all supplementary CNN models with a precision of 94%. However, when we took both types of inputs simultaneously, the accuracy for detection was increased to 99.7%. After extensive experimentation, we believe that our proposed hybrid method is robust enough to take X-rays and audio as mel-spectrograms and identify COVID-19 at early stages, attaining an accuracy of 99.7%.

Keywords CNN · CT scan · X-rays · COVID-19 · Deep learning · HCI

1 Introduction

In December 2019, numerous cases of pneumonia were identified in Wuhan. SARS-Cov-2 is an acute respiratory syndrome that affects breathing severely. This epidemic killed millions of people worldwide and it spreads from one person to another very easily. According to WHO, on March 30, 2022, there were 483,556,595 verified COVID-19 cases, with 6,132,461 fatalities documented [1]. Cough, fever, muscular or body pains, shortness of breath, headache, tastelessness, diarrhea, and sore throat characterize COVID-19. COVID spreads through drops and infection particles when a tainted individual makes coughs or wheezes. Therefore, necessary precautions include keeping a social distance, wearing a mask, and washing hands [2]. The incubation period for coronavirus patients is 3–12 days.

✉ Rabbia Mahum
rabbia.mahum@uettaxila.edu.pk

✉ Haseeb Hassan
haseeb@sztu.edu.cn

Farooq Akhtar
farooq.akhtar@students.uettaxila.edu.pk

Adham E. Ragab
aragab@ksu.edu.sa

Faisal Shafique Butt
faisalbutt@ciitwah.edu.pk

Mohammed A. El-Meligy
mohammedali2000@gmail.com.sa

[1] Department of Computer Science, University of Engineering and Technology, Taxila, Pakistan

[2] Industrial Engineering Department, College of Engineering, King Saud University, PO Box 800, 11421 Riyadh, Saudi Arabia

[3] Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt, Pakistan

[4] Jadara University Research Center, Jordan University, Jordan, Jordan

[5] Applied Science Research Center, Applied Science Private University, Amman, Jordan

[6] College of Big Data and Internet, Shenzhen Technology University (SZTU), Shenzhen, China

Published online: 10 September 2024

There exist several manual methods for the detection of COVID-19 virus, including polymerase chain reaction (RT-PCR) and lung screening, such as X-rays and CT images, which are utilized to detect the infection. Hence, RT-PCR test is not a solid marker for the presence of COVID-19 disease because it requires a significant amount of resources, manual effort, substantial costs, and considerable time investment. Furthermore, performing an RT-PCR test mandates the involvement of proficient personnel who are well-trained in utilizing RT-PCR kit to carry out the test, explicitly using throat or nasal swabs for detecting SARS-CoV-2. Additionally, a comprehensive arrangement encompassing skilled professionals, a laboratory, and an RT-PCR apparatus is imperative for conducting RT-PCR tests [3, 4]. Therefore, the researchers have focused on various automated systems that can detect the presence of infection and ways to control it at an early stage.

CT and X-ray scans are two ways that can aid in the diagnosis of pandemics in their early stages. The chest X-ray (CXR) imaging method is generally favored over CT scans due to the latter's elevated radiation dosage, which is particularly concerning for pregnant women and children. In contrast, CXR employs minimal radiation, reduces cross-infection risk, and is extensively accessible compared to CT scans. Nevertheless, numerous limitations exist associated with manually diagnosing COVID-19 using CXRs. For example, it consumes more time and is susceptible to humanoid mistakes. Likewise, the pandemic situation demands a substantial number of radiologists for the timely diagnosis of COVID-19 using CXRs manually. Hence, there is a need for an automated approach to achieve precise COVID-19 diagnosis. In recent times, several approaches based on deep learning (DL) have been introduced to accomplish this objective.

Artificial intelligence (AI) is a wide term that includes several subcategories, such as Machine Learning (ML) and Deep Learning (DL) [5–8]. The DL-driven methods are introduced in recent studies for COVID-19 diagnosis [9, 10]. They are categorized into two primary groups: segmentation-based strategies and classification-based approaches. The classification-based approaches mostly utilized convolutional neural networks (CNN), such as ResNet, DenseNet, and VGG, which extract the overall features from lung images to recognize COVID-19 infection. On the other hand, segmentation-based methodologies focus on analyzing the regions affected by COVID-19 in the lungs. This is accomplished by the network's training to identify the area of lung, after which the segmented region is fed into the network for classification [11]. However, segmentation-based techniques still encounter certain constraints. For instance, these methods can be overly sensitive to intricate shapes in the impacted regions. Additionally, their performance is heavily reliant on meticulously annotated training data, which is a tiring process.

However, DL and ML models [10, 11] have significant drawbacks. These include their reliance on large datasets for training, the necessity of expensive computational resources like graphical processing units (GPUs), extensive trainable parameters, large feature vector sizes, and prolonged running, training, and testing times. On the other hand, transfer learning models have raised concerns regarding negative transfer and overfitting. Therefore, in this work, to overcome the challenge of a large dataset, a hybrid technique is presented for the detection of COVID-19 that can notice a minor infection in chest images with maximum precision and accuracy using images and audio as well. Main features of this study are below:

- To introduce an efficient framework for early detection of COVID-19 based on the models of DL.
- Two models are proposed i.e., speech-based and image-based. Speech-based model is using VGGish that converts audios into mel-spectrograms. Whereas, image-based model that is ResNet50, is selected after analyzing several DL techniques.
- The audio-based model proposed in this study demonstrates remarkable robustness, achieving 98.9% testing accuracy, while ResNet50 provides 94% accuracy.
- For effective detection, the powers of audio and image-based models are combined to form a hybrid approach that takes the input from audio and X-rays simultaneously, making a detector more robust.
- The hybrid system, comprising image and audio-based, performs significantly for COVID-19 detection, achieving 99.7% accuracy and outperforming the existing models.
- Several experiments have been performed to ensure the outperformance of the proposed hybrid framework for COVID-19 detection.

Section 2 explains related work, Sect. 3 displays two proposed techniques, audio-based and picture-based, Sect. 4 describes the experimental section, and the last section describes the conclusion.

2 Related Work

Alsabek et al. suggested a model for extracting Mel-frequency cepstral coefficients (MFCCs) and audio digital information from negative and positive COVID-19 patients and generating coefficients [2]. They suggested a cost-effective approach for extracting data from non-COVID and COVID patients that incorporates MFCCs and speech signal processing, and generated personal correlations from their relationship coefficients. Hassan et al. [8] built a COVID-19 classification model employing long short-term

memory (LSTM) to assess the acoustic characteristics of the patients' cough, speech, and breathing. They used audio data that involved together COVID-19 and non-COVID-19 voice examples. When compared to coughs and breath sounds, with 98.20% and 97.0% accuracy, respectively, the voice test had a low accuracy of 88.20%.

Researchers, physicians, and scientists from all across the world have been paying close attention to the COVID-19 pandemic forecasts. The authors in [12] proposed a COVID-19 detector that was simple to use. To distinguish between positive and negative coughs, their screening system captured coughs using a cell phone and applied numerous DL algorithms. They used two datasets that included both natural and induced coughs collected from six continents. The Coswara dataset comprises 1080 healthy samples and 92 samples of COVID-19-positive patients. The other dataset was obtained in South Africa and included 21 audio from 8 positive COVID-19 and 13 COVID-19-negative people. MLP, logistic regression, LSTM, SVM, CNN, and RNN were employed in the study. Among the classifiers, the Resnet-50 produced good results, attaining an accuracy of 94.5%.

In the study [13], an automated COVID-19 identification system showed how acoustic patterns of coughs may be used to detect the diagnosis. The researchers used an encoder–decoder mechanism in which the encoder encrypted the breathing patterns from audio signals, and the decoder used an attention mechanism to decode the state of COVID-19 for the correct detection. To attain a 64.42% area under the curve, the encoder employed a layered BI-LSTM network. Muhammad Haris Munir et al. [14] recommended a model for the identification of COVID-19 built on CT scans and chest X-rays. The researchers utilized a DL model by combining Alex-Net and Faster RCNN to identify corona virus. The model's average accuracy was 98.3%, which indicated the model's efficiency.

The authors in [15] suggested a method for detecting patients with COVID-19 based on cough noises. The researchers used a long standing transformation, an equal rectangular bandwidth (ERB) distribution, and a gammatone filter bank to create a unique audio signal. The CQT spectrum, the GTCC, and the ERB were used. Additionally, the MFCCs were assessed using the LR, RF, and MLP techniques. The combined form of ERB spec-RF attained the peak of 81.89% AUC term.

To detect COVID-19 symptoms, Jord et al. [16] projected a model of DL based on cough gathered by mobile phones. This approach allowed a low-cost option for pre-screening of COVID-19 all over the world. Cough recordings were processed using MFCCs and fed into a CNN-based architecture with one network layer and 3 ResNet50s running in parallel, yielding a binary pre-screening diagnosis. For the random 5320 samples, the model predicted COVID-19-positive symptoms with 97.1% accuracy and detected asymptomatic COVID-19 with 100% accuracy. Table 1 lists the state-of-the-art work along with their results.

3 Methodology

In this study, we are proposing two separate models for COVID-19 detection: a speech-based classifier and an image-based classifier. In the first model, we utilized the speech signals (coughing and breathing) of various COVID-19-infected and healthy people. Furthermore, we transformed these audio signals into mel-spectrograms and trained an improved VGGish network for binary classification into COVID-19 and non-COVID patients. In the second model, which is based on images, we employed various networks; however, ResNet50 found to be better than others in terms of accuracy. The details of both models are presented below.

3.1 Speech-Based Model

3.1.1 Data Acquisition

The training step of deep neural networks necessitates an adequate amount of data. Therefore, the first step of this method was data gathering. We collected the speech data consisting of 1200 sound samples from 600 people, including patients and healthy persons. The samples in the speech corpus were collected from a hospital in Pakistan. The samples exhibited breathing, coughing, and speaking sounds. Among 600 participants, 350 persons were healthy, whereas 250 were COVID-19 patients. The data were gathered using Android mobile. All the participants were guided to take a deep breath while counting from 1 to 10 and coughing four times. Furthermore, all the patients were asked to sit in a relaxed position during the speech recording. Each speech was recorded thrice to overcome the quality challenge of the mobile microphone. Additionally, data were obtained from the open-source Coswara database.

3.1.2 Our Customized Network

In this section, our proposed network, i.e., an improved VGGish [22] based on the VGG for audio classification, is explained. The building block of the suggested approach is shown in Fig. 1. VGG is a convolutional neural network having 3×3 filters. Other than its simplicity, the network is distinguished by its inclusion of pooling layers and fully connected layers.

Consider a basic fully connected layer, where the output column vector is computed by:

$$y^t = [y_1, y_2, \ldots, y_n], \tag{1}$$

Table 1 Existing techniques and results from previous work

Works	Techniques	Description	Performance measures
Alsabek et al. [2]	MFCC features	COVID-19 identification is performed by analyzing MFCC acoustic features and providing correlation coefficient evaluation	The average linear relationship is 0.42
Hassan Abdelfatah et al. [9]	LSTM	COVID-19 is diagnosed early, and many acoustic aspects are evaluated	Accuracy: 98.2% AUC: 98.8%
Pahar Madhurananda et al. [12]	RSNET, MLP, LR, LSTM, SVM, CNN, and residual-based neural network	Identifying positive and negative coughing results	Accuracy: 95.34% AUC: 97.5%
Deshpande et al. [13]	BI-LSTM	Cough analysis allows for early identification of COVID-19	AUC: 64.43%
Kumar et al. [15]	LR, FR, MLP	Cough analysis is being used to provide an early screening for COVID-19	AUC: 81.89%
Laguarta Jord et al. [16]	CNN, MFCC, Resnet-50	Cough analysis is used to provide an advanced screening for COVID-19	AUC: 97.3%
Gunavant et al. [17]	MFCC	Cough analysis will be used to provide an early preview for COVID-19	AUC: 77.1% ROC: 77.1% Accuracy: 78.3%
Maghdid et al.[18]	Alex-Net and CNN	Identification of COVID-19 using patient X-rays and CT pictures	AUC: 98%
Wang et al. [19]	DensNet, Nasnet-Amobile, Resnet-50	Sensing the existence of COVID-19 using chest CT imaging	AUC: 99.1%
Jaiswal et al. [20]	CNN: Densenet-201	CT imaging is being used to identify the presence of COVID-19	Accuracy: 97%
Weng et al. [19]	Inception pre-trained CNN	Identification of corona virus using CT scan images	Accuracy: 89.5% Specificity: 88% Sensitivity: 87%
Narin, Ali et al. [21]	ResNet152, InceptionV3, ResNet50, and Inception-ResNetV2	Chest X-ray radiography can identify coronavirus individuals	AUC: 99.7%

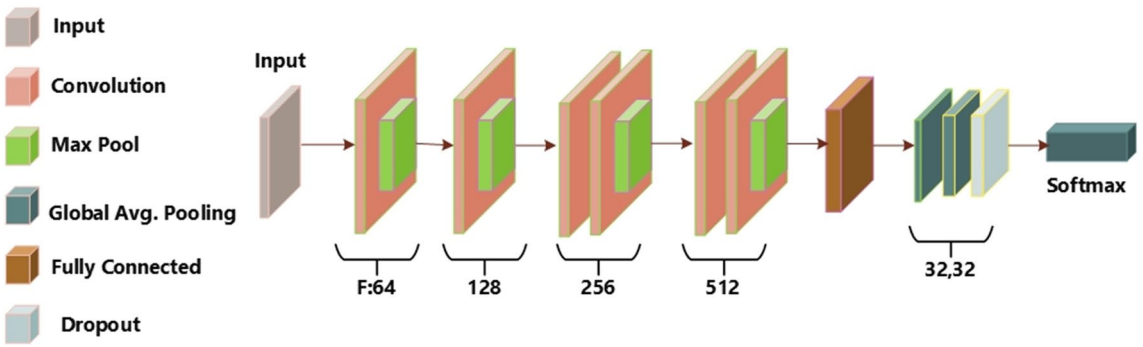

Fig. 1 Architecture of the VGGish model

$$y = f(Wx + b)(1), \tag{2}$$

$$x^t = [x_1, x_2, \ldots\ldots, x_m], \tag{3}$$

$$b^t = [b_1, b_2, \ldots.., b_n], \tag{4}$$

where f is the active function, matrix W presents weights, b is biased, and x is the input unit vector. The input vector x and trainable parameters (matrix W) are multiplied to form a linear mapping in the first layer. This matrix is added by the bias vector b and the activation function f. The second component performs a non-linear mapping to produce the output vector y.

The existing VGGish model consists of 26 layers for audio classification [23], which have convolutional, max-pooling, and ReLU layers. However, for the COVID-19 detection, we have improved the original architecture, as shown in Fig. 4. Leaky ReLU [24] was introduced by "giving a modest negative gradient for negative inputs" instead of the ReLU function rather than being 0. When compared to the ReLU activation function, this modification of Leaky ReLU can result in minor gains in classification performance. Moreover, Leaky ReLU overcomes the issue of vanishing gradient better than the ReLU, therefore helps in effective training. Traditional convolution layers utilize two phases. The input feature maps are convolved first. Then, activation functions are used to build output feature maps in the second stage. The architecture of the suggested model is revealed in Table 2.

At first, all the input audio files were converted into mel-spectrograms, which present the frequencies in the form of mel-scale. The reason behind this conversion was the greater effectiveness of the proposed network in extracting visual patterns. It can be seen from Table 2 that an input image with dimensions of 96×64 is fed to an input layer with one channel of information. Then, 3×3 convolution is used on the input. After the convolutional operation, Leaky ReLU is used for the initiation of the features coming from the convolutional layer. Moreover, the max-pooling layer has been employed with stride 2 to reduce the dimensions of the input coming from an activation layer.

Then, the convolutional layer uses the filter size of 128, having stride 1 and the same padding. We used the Leaky ReLU layer with a scale of 0.1 to utilize the features coming from the convolutional layer. In the next step, the max-pooling layer reduces the dimensions with stride 2. Furthermore, there is a convolutional layer with 3×3 filter using stride 1. After the convolutional procedure, the features from the convolutional layer are activated using the Leaky ReLU activation function. In the next step, we used two convolutional layers using 256 filters with 3×3 convolutions and a stride of 1 with the same padding. Then, we employed leaky ReLU with a 0.1 scale. A max-pooling layer with size of 2×2 and a stride of 2 is used. Further, there is a convolutional layer with 3×3 filter with stride 1. After the convolutional procedure, the features from the convolutional layer are activated using the Leaky ReLU activation function. In the next step, we utilized a convolutional layer with 512 filters, 3×3 convolutions, and a stride of 1 with the same padding. Then, we utilized Leaky ReLU with a 0.1 scale along with Max-pooling with a pool size of 2×2 and a stride of 2 is used. Moreover, a fully connected layer with neurons of 4096 has been employed. Then, we employ a leaky ReLU as an activation function, and further, these two layers are repeated twice. Thus, a layer of convolution with a 1×1 filter that is 1000 in size using stride 1 is utilized. After that, another additional layer of average global pooling is utilized in our proposed model for down sampling by computing the mean of the height and width of the input. Then, a fully connected layer with an output size of 2 is utilized. In the end, for the classification, the softmax layer for normalizing the input data has been employed along with the classification layer.

Algorithm for the speech-based model

Input: Audio samples (breathing, coughing, speaking)
Output: Classified Audios as Patient or Non-Patient
Start:
1. $[A_{train}, A_{test}] \leftarrow$ *Split Audios*
2. *Pro_Audios* $\leftarrow$ *Resampling(16000Hz,* A_{train} *)*
3. $€ \leftarrow$ *Bark-Spectrum(Pro_Audios)*
4. *Ms* $\leftarrow$ *Mel_spec (Image_size,* $€$*)* // Image_size=96 x 64
5. Start Training of VGGish:
6. For $\forall$ Ms *x* in $\rightarrow A_{train}$
   a) *Image Input layer*
   b) Features extraction
   c) Classification
   End For
7. £$\leftarrow$Trained Model VGGish

8. WHILE$\forall x \in A_{test}$
   1.Resampling(16000Hz)
   2.ή$\leftarrow$Conversion (96 x 64)
   2. *Features*$\leftarrow$ ή
   3. Classification through trained classifier £

9.End While
10. *Accuracy* computation for Evaluation of Model.
End

Table 2 Architecture details of the proposed network

	Layer	Type	Activations	Learnable
1	Image input 96×64×1 images	Image input	96×64×1	–
2	Conv1 64 3×3 convolutions with stride [1] and padding 'same'	Convolution	96×64×64	Weight 3×3×1×64 Bias 1×1×64
3	Leaky Leaky ReLU with a scale of 0.1	Leaky ReLU	96×64×64	–
4	Pool 1 2×2 max-pooling with stride [2×2] and padding 'same'	Max-pooling	48×32×64	–
5	Conv 2 128 3×3convolutions with stride [1] and padding 'same'	Convolution	48×32×128	Weight 3×3×64×128 Bias 1×1×128
6	Leaky1 Leaky ReLU with a scale of 0.1	Leaky ReLU	48×32×128	–
7	Pool 2 2×2 max-pooling with stride [2×2] and padding 'same'	max-pooling	24×16×128	–
8	Conv 3_1 256 3×3 convolutions with stride [1] and padding 'same'	Convolution	24×16×128	Weight 3×3×128×256 Bias 1×1×256
9	Leaky2 Leaky ReLU with a scale of 0.1	Leaky ReLU	24×16×128	–
10	Conv 3_2 256 3×3 convolutions with stride [1] and padding 'same'	Convolution	24×16×256	Weight 3×3×256×256 Bias 1×1×256
11	Leaky3 Leaky ReLU with a scale of 0.1	Leaky ReLU	24×16×256	–
12	Pool 3 2×2 max-pooling with stride [2×2] and padding 'same'	max-pooling	12×8×256	–
13	Conv 4_1 512 3×3 convolutions with stride [1] and padding 'same'	Convolution	12×8×512	Weight 3×3×256×512 Bias 1×1×512
14	Leaky4 Leaky ReLU with a scale of 0.1	Leaky ReLU	12×8×512	–
15	Conv 4_2 512 3×3 convolutions with stride [1] and padding 'same'	Convolution	12×8×512	Weight 3×3×512×512 Bias 1×1×512
16	Leaky5 Leaky ReLU with a scale of 0.1	Leaky ReLU	12×8×512	–
17	Pool 4 2×2 max-pooling with stride [2×2] and padding 'same'	Max-pooling	6×4×512	–
18	Fc1_1 4096 fully connected layer	Fully connected	1×1×4096	Weight 4096×12,288 Bias 4096×1
19	Leaky6 Leaky ReLU with a scale of 0.1	Leaky ReLU	1×1×4096	–
20	Fc1_2 4096 fully connected layer	fully connected	1×1×4096	Weight 3×3×1 Bias 1×1×64
21	Leak7 Leaky ReLU with a scale of 0.1	Leaky ReLU	1×1×4096	–
22	Fc1_3 4096 fully connected layer	fully connected	1×1×4096	Weight 4096×4096 Bias 4096×1
23	Leaky8 Leaky ReLU with a scale of 0.1	Leaky ReLU	1×1×4096	–
24	Conv 1000 1×1 convolutions with stride [1] and padding 'same'	Convolution	1×1×1000	Weight 1×1×4096×1000 Bias 1×1×1000
25	Avg1 Global average pooling	Global avg. pool	1×1×1000	–
26	Fully connected layer	FC	1×1×2	Weight 2×1000 Bias 2×1
27	SoftMax	SoftMax	1×1×2	–
28	Classification	Classification output	1×1×2	–

3.2 Images-Based Model

In this section, we describe the details of image-based techniques for COVID-19 detection.

3.2.1 Data Collection

The dataset contains 6557 images gathered from Kaggle [25]. The various images of infected and uninfected cells from the dataset have been selected arbitrarily. The X-ray images of the chest (anterior–posterior) were selected from historical datasets containing pediatric (1–5 years), and matured patients obtained from the Guangzhou Ladies and Kids' Clinical Center in Guangzhou. These chest X-beam images were a normal part of the patients' clinical consideration. To ensure the reliability of the chest X-ray analysis, an initial screening process was employed to eliminate all scans that exhibited low quality or were unreadable. Subsequently, two expert physicians assessed and graded the diagnoses associated with these images before they were considered suitable for the model's training. To mitigate the potential for grading inaccuracies, a third expert independently reviewed the evaluation set as well.

3.2.2 Preprocessing

Image preprocessing was used to eliminate abnormalities and improve image properties. Image preparation included scaling and resizing to provide images of the same size as the model takes. The images originally had varying dimensions; therefore, they were resized to 227×227 for the experimental setting. Furthermore, data augmentation was utilized to address the issue of limited training samples. The major objective was to enlarge the training dataset. The data augmentation method included brightening tweaking, expanding and resizing, twisting, and vertical or horizontal flipping. During the training phase, numerous types of transformations and distortions are often used, none of which affects the semantics of the images [26, 27].

3.2.3 CNNs

Eight distinct CNN models were used in this study, including VGG16, DenseNet201, ResNet50, InceptionV3, squeezeNet, DarkNet, GoogleNet, and Alex-Net. VGGNet is regarded as the most noteworthy and widely used architecture [28, 29]. The VGGNet architecture has 16 to 19 convolution layers, three filters, five max-pooling, three fully linked layers, and a classification layer. ResNet, like other feed-forward networks, has residual connections. The residual unit's ultimate output is $\times 1$ and can be calculated using the subsequent equation [30]:

$$xl = F(xl-1) + xl-1, \tag{5}$$

The Xception structure consists of convolutional operations that are stacked and divided depth-wise. It contains 36 layers that serve as the network's feature extraction foundation [31]. The DenseNet is composed of densely connected layers with outputs interconnected to all of their successors in a dense block [32]. Densnet201 contains 201 layers loaded with images' weights. Furthermore, InceptionV3 is used to improve processing resources by increasing the internal layers of the network [33]. The architecture is made up of 48 network layers. To minimize dimensionality, the suggested architecture employs max-pooling [34].

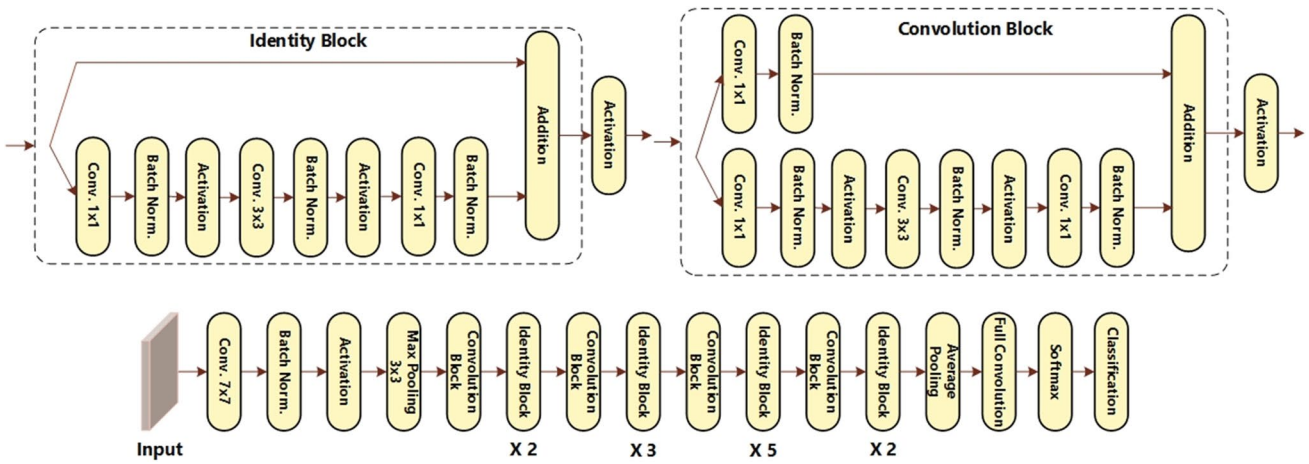

Fig. 2 ResNet50's architecture with internal layer-wise detail

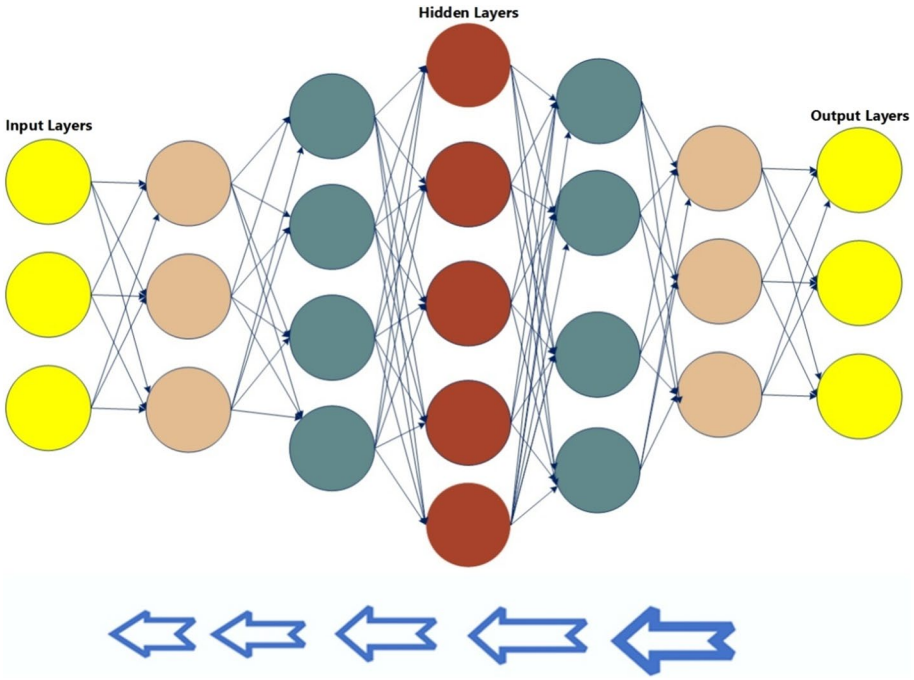

Fig. 3 Vanishing gradients in CNN

3.2.4 Our Proposed Model

We chose a Residual Network (ResNet50) for the reliable performance of the model. The acronym ResNet50 stands for Residual Network with 50 layers. The residual network is made up of numerous types of residual blocks. The processes inside the residual block, on the other hand, vary depending on the design of the residual networks. Figure 2 depicts the ResNet50 architecture's core block design.

The assumption is that the more deeply the network gets in, the improved classification results are [35]. When the neural network is excessively deep, the gradient value decreases to 0, causing the weights to stop getting updated, and no learning occurs. The phenomenon of vanishing gradients is seen in Fig. 3. Deep networks encountered several challenges, including network optimization, deterioration, and, most crucially, disappearing gradients. According to the research, fine-tuning a pre-trained CNN network can enhance the accuracy in the corresponding domain [36, 37].

Transfer learning is an approach that allows one to get training in one domain and re-purpose it in another area [38]. Two important philosophies of transfer learning exist, task and domain, and they are mathematically explained in [39]. A domain D is made up of two components: a feature space θ and a marginal distribution function $P\ (F)$.

$$D = \{\theta, P(F)\}, \tag{6}$$

In this instance, F stands for an instance defined as $F = \{\ x|x_i \in= \theta, i = 1, \ldots .,\}$. A task T depends on a label space L, and a judgment function t as below:

$$T = \{L, t\}, \tag{7}$$

"Network surgery" was utilized for fine-tuning the deep neural network (DNN), i.e., ResNet50. The "fc1000 softmax" and "Classification Layer fc1000" layers have been eliminated from the network. The new layers were added to replace these. A network head was created using the additional architectural layers. Three layers made up the network's head, including a fully connected layer with a value of 20 for both the WeightLearnRateFactor and the BiasLearnRateFactor. A new softmax layer was the second layer, and the last layer added to the network head was a new classification layer.

4 Experiments and Results

4.1 Grid Search

Before training, hyper-parameters are initiated in a machine-learning model. These hyper-parameters need to be tuned for a model to be applicable to a dataset. Nonetheless, the optimal variable settings on a specific dataset are unlikely to be excellent on another, making feature optimization unachievable. Grid search is a traditional hyper-parameter optimization technique that guarantees that the search is limited to a

specified subsection of the training method's hyper-parameter space. The algorithm then starts a full search over these parameters. The following steps were carried out:

- Selection of hyper-parameters to be tuned.
- Setting the range for each hyper-parameter.
- Obtaining all combinations systematically.
- During model training, each combination of hyper-parameters is evaluated based on root mean square error on root values and computational time, with the stopping condition being the number of epochs.
- Ranked the hyper-parameters combinations first by the least root mean squared values, then by computational time, breaking ties arbitrarily.

The following ranges were selected for both proposed models.

Learning rate: [0.01, 0.001, 0.001].
Epochs: [50, 100].
Dropout: [0.1, 0.3, 0.5].

4.2 K-Fold Cross-Validation

The cross-validation approach was used to test the performance after the models had been brute-forced [40]. Cross-validation is a widespread method for assessing models' actual estimation errors and modifying the parameters of the models to avoid false predictions [41]. This elegant strategy is extensively utilized to address the overfitting problem that several methods have as a result of dataset irregularity (small size) [42, 43]. Before using the K-fold cross-validation, the data training was separated into K pieces, each of which comprised an n/k sample, where n presents the total number of training occurrences. Hence, only the first k-1 sections were utilized in learning, while the remaining portions were used in validation. Additionally, generalization problems were addressed using the K-fold cross-validation method. The parameters utilized during the k-fold cross-validation are given in Table 3.

Table 3 The statistics for K-fold cross-validation

Parameter	Description
Number of folds	10
Stratification	Enabled: (due to imbalanced training data)
Shuffle	Enabled
Random seed	42
Hyperparameter tuning	Independent to avoid overfitting
Data leakage	Precautions are taken to avoid data leakage

4.3 Metrics

Accuracy, F1 score, recall, and precision are metrics that have been used to assess the performance. The four components of the metrics are provided below:

- True positive (TP)
- False positive (FP)
- False negative (FN)
- True negative (TN)

The following equation can be utilized to compute Accuracy:

$$\text{Accuracy} = (\text{TP} + \text{TN})/(\text{TP} + \text{FP} + \text{TN} + \text{FN}), \tag{8}$$

In terms of all (positive) predictions, Precision is the percentage of accurately predicted (positive) class samples. The Precision is determined using the formula shown below:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{9}$$

The proportion of accurately estimated (positive) predictions that the model recognizes is known as recall. The following equation is applied to compute the Recall:

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}), \tag{10}$$

The F1 score denotes the balance between Accuracy and Recall. The F1 score equation is given below:

$$\text{F1} - \text{score} = 2 \times ((\text{Recall} \times \text{precision})/(\text{Recall} + \text{precision})), \tag{11}$$

4.4 Speech-based Model's Results

The proposed speech-based network makes use of the VGGish model. The three components of the speech dataset are training, testing, and validation. 30% of the speech dataset was utilized for testing, 10% for validation, and 70% for training. TensorFlow served as the deep learning framework. To boost the data and improve the system's accuracy, data augmentation was employed.

The achieved outcomes over the audio dataset for diagnosing COVID-19 are given in Table 4. We performed several experiments by selecting varying learning rates for 50 and 100 epochs. Further, we selected 3 Batch sizes: 16, 32, and 64 for both scenarios. Similarly, the dropout factor was varying as 0.1, 0.3, and 0.5 for each batch size. Among these different hyper-parameters, the best results were attained on 100 epochs, batch size of 32, demit rate of 0.1, and rate of learning as 0.001. On the other side, for 50 epochs, we attained second highest accuracy for batch size 32, dropout rate of 0.1, and learning rate of

Table 4 Performance of a speech-based model for various hyper-parameter values

Epochs	Learning rate	Batch size	Dropout	Accuracy (%)	F1 score (%)	Precision (%)	Recall (%)
50	0.01	16	0.1	74.80	77.40	78.20	74.80
	0.001			93.30	93.40	95.60	93.30
	0.0001			87.60	88.50	89.40	87.60
	0.01		0.3	71.10	74.20	77.60	71.10
	0.001			94.20	94.60	95.00	94.20
	0.0001			82.20	83.10	84.10	82.20
	0.01		0.5	61.10	66.30	72.50	61.10
	0.001			91.60	91.60	91.60	91.60
	0.0001			76.20	78.00	79.80	76.20
	0.01	32	0.1	85.90	86.40	86.90	85.90
	0.001			98.40	98.40	98.40	96.40
	0.0001			83.30	84.40	85.50	83.30
	0.01		0.3	72.50	75.20	78.00	72.50
	0.001			96.20	96.20	96.20	96.20
	0.0001			79.60	81.70	83.90	79.60
	0.01		0.5	68.50	71.90	75.70	68.50
	0.001			93.90	94.40	94.90	93.90
	0.0001			66.00	68.40	71.10	66.00
	0.01	64	0.1	92.70	93.40	94.10	92.70
	0.001			92.40	92.70	93.00	92.40
	0.0001			74.20	76.80	79.60	74.20
	0.01		0.3	78.20	79.60	81.00	78.20
	0.001			97.30	97.60	95.80	97.30
	0.0001			65.40	70.90	74.70	65.40
	0.01		0.5	69.70	72.90	76.50	69.70
	0.001			93.00	93.20	93.30	93.00
	0.0001			62.80	67.60	73.10	62.80
	0.01	16	0.1	83.30	85.40	87.60	83.30
	0.001			98.60	99.60	99.60	99.60
	0.0001			92.60	91.70	91.90	91.60
	0.01		0.3	76.90	80.50	83.30	77.90
100	0.001			95.40	94.70	95.00	94.40
	0.0001			93.20	93.30	94.40	94.20
	0.01		0.5	77.20	80.10	82.10	76.20
	0.001			94.30	94.30	95.30	95.30
	0.0001			88.30	88.60	87.30	87.30

Table 4 (continued)

Epochs	Learning rate	Batch size	Dropout	Accuracy (%)	F1 score (%)	Precision (%)	Recall (%)
	0.01	**32**	**0.1**	91.50	92.40	94.30	91.50
	0.001			**98.90**	**98.90**	**97.90**	**98.90**
	0.0001			84.90	85.40	86.90	84.90
	0.01		0.3	76.10	78.00	78.90	76.10
	0.001			98.00	98.00	97.00	98.00
	0.0001			94.30	94.40	93.60	94.30
	0.01		0.5	87.50	88.20	89.00	87.50
	0.001			93.40	93.70	94.00	93.40
	0.0001			89.20	89.60	89.90	89.20
	0.01	64	0.1	90.30	90.70	91.10	90.30
	0.001			96.00	96.00	96.00	96.00
	0.0001			90.90	91.30	91.70	90.90
	0.01		0.3	87.50	88.30	89.20	87.50
	0.001			97.40	97.40	97.40	97.40
	0.0001			90.30	90.70	91.10	90.30
	0.01		0.5	59.80	65.10	71.40	59.80
	0.001			96.90	97.10	97.10	96.90
	0.0001			86.90	87.60	88.40	86.90

Bold values are selected for training the model

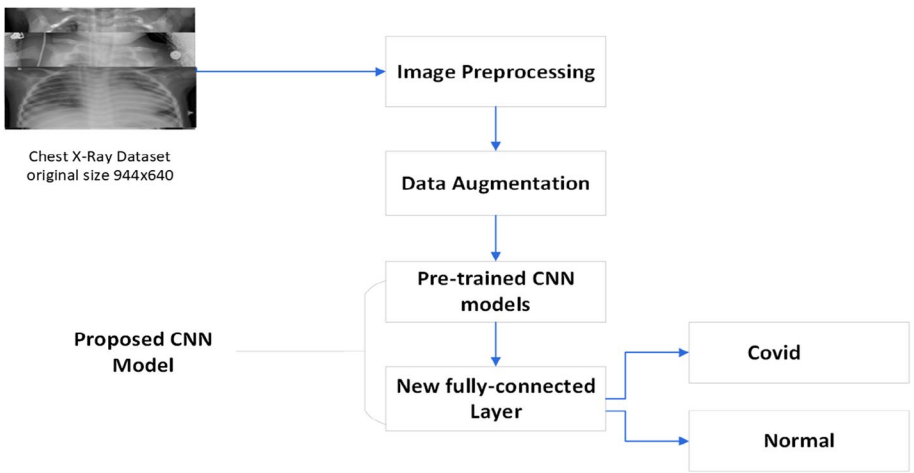

Fig. 4 Image-based COVID-19 detector's architecture

0.001. More specifically, the ideal parameters in both scenarios are same. Therefore, the results ensured that these hyper-parameters must be selected with 100 epochs for final classification.

4.5 Results of Image-based Classifier

The suggested CNN model utilized batch size: 32, dropout: 0.1, epochs: 50, and the rate of learning: 0.001, employing the grid search optimization approach. Additionally, using the augmentation technique, the performance of the proposed model was enhanced. In addition, generalization problems were addressed using the K-fold cross-validation. Figure 4 shows the steps utilized to develop the COVID-19 approach that relies on the X-rays. To identify the existence of COVID-19 in this framework, binary classification was performed. Further, eight different CNN models, including VGG16, Alex-Net, ResNet50, DenseNet201, InceptionV3, DarkNet, SqueezeNet, and GoogleNet, were used in the experiment. Figure 5 illustrates how the Resnet-50 model performs better than all others, with a maximum accuracy of 94.5%.

The performance measures of the proposed image-based model are reported in Table 5. It is clearly visible that the results are in decreasing order from top to bottom. The SqueezeNet and Alex-Net attained the worst performance. The reason could be the limited capacity of feature extraction by SqueezeNet, and the simple fixed architecture of Alex-Net. Moreover, GoogleNet attained 78.6% that is not much considerable. After that DenseNet201, DarkNet, InceptionNetV3, and VGG16 provided almost similar outcomes; however, we could not compromise for the detection accuracy of COVID-19 due to its life threating effects. Therefore, we chose ResNet50 for the image-based classification due to its better performance for image classification tasks. It is notable that we attained maximum accuracy by ResNet50.

The performance for several hyper-parameters by image-based classifier can be seen in Table 6 below.

Table 5 The performance of the proposed image-based classifier

Classifier	Accuracy (%)	Precision (%)	Recall (%)
ResNet50	**94**	**93.4**	**94.7**
VGG16	85.6	83.3	82.4
InceptionV3	84.4	83.2	84.5
DarkNet	81.6	79.2	80.7
DenseNet201	81.3	80.3	78.2
GoogleNet	78.6	80.2	81.3
Alex-Net	58.5	59.2	62.3
SqueezeNet	56.4	58.3	59.2

Bold values are selected for training the model

4.6 Hybrid Model's Performance

In this experiment, we combined the performance of two proposed models: image-based and audio-based that accept both inputs i.e., audio in the form of mel-spectrograms and X-rays. The steps followed in the COVID-19 recognition system are shown in Fig. 6. To indicate the existence of COVID-19 infection, the system was trained over binary classes. When compared to the effectiveness of the only image and speech-based systems with the hybrid model, the efficiency of both solo systems was significantly inferior. The image-based system achieved 94%, the speech-based system attained 98.90%, whereas the hybrid model attained 99.7% accuracy. The results are attained following the strategy as shown in Table 7. Both pre-trained networks are assisting in the detection of COVID-19 due to the combined powers of VGGish and ResNet50. The difference of 0.8% is due to the inclusion of a speech-based model. As a result, we can infer that our suggested hybrid model is sufficiently robust to identify COVID-19 effectively for both types, such as speech and X-rays. The reason could be the pre-trained VGGish network that has the properties of an image classification algorithm, i.e., VGG and audio classification task. Further, ResNet50 is already known for better image classification tasks.

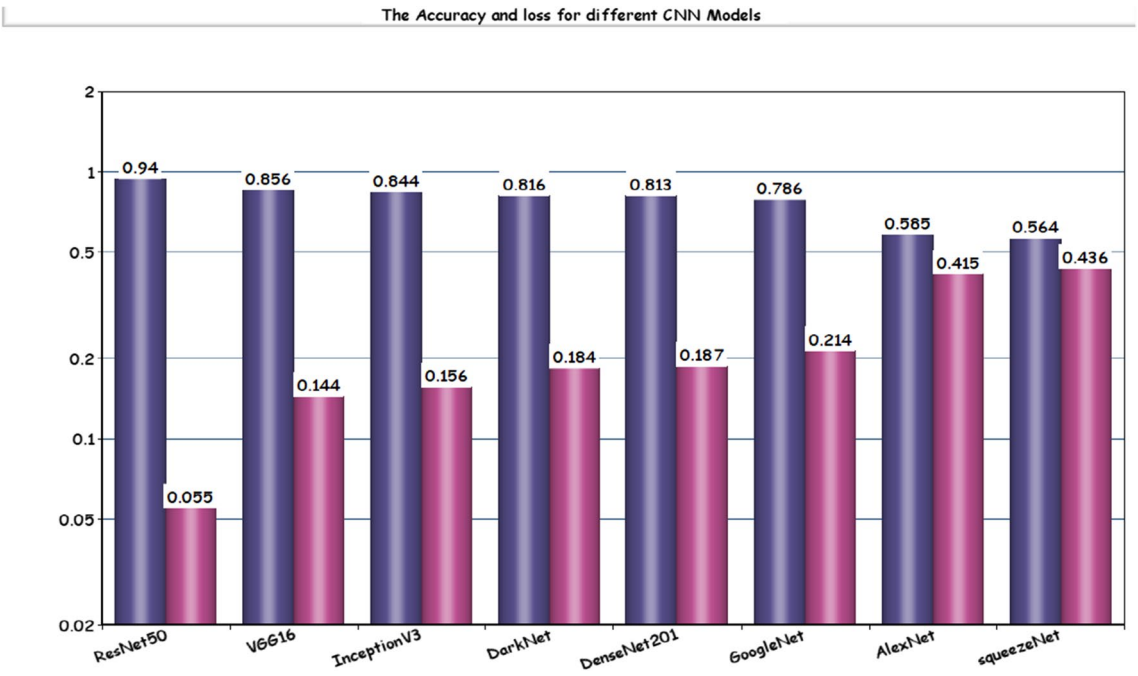

Fig. 5 The accuracy and loss comparison for different CNN models

Table 6 The performance of the proposed image-based classifier on varying hyper-parameters

Epochs	Batch size	Learning rate	Dropout	Accuracy (%)	Precision (%)	Recall (%)
50	16	0.01	0.1	85.80	88.8	89.3
		0.001		53.30	56.3	56.8
		0.0001		77.60	80.6	81.1
		0.01	0.3	81.10	84.1	84.6
		0.001		84.20	87.2	87.7
		0.0001		82.20	85.2	85.7
		0.01	0.5	71.10	74.1	74.6
		0.001		69.60	72.6	73.1
		0.0001		76.20	79.2	79.7
	32	0.01	**0.1**	88.30	91.3	91.8
		0.001		**94**	**93.4**	**94.7**
		0.0001		73.20	76.2	76.7
		0.01	0.3	72.50	75.5	76
		0.001		91.20	90.2	90.7
		0.0001		79.99	82.99	83.49
		0.01	0.5	78.50	81.5	82
		0.001		83.90	86.9	87.4
		0.0001		76.00	79	79.5
	64	0.01	0.1	92.70	91.7	90.2
		0.001		90.40	89.1	97.9
		0.0001		78.20	81.2	81.7
		0.01	0.3	79.20	82.2	82.7
		0.001		77.30	80.3	80.8
		0.0001		75.40	78.4	78.9
		0.01	0.5	79.70	82.7	83.2
		0.001		92.01	91.01	95.51
		0.0001		72.80	75.8	76.3

Bold values are selected for training the model

4.7 Comparison with Existing Techniques

We compare the outcomes of our proposed model with the existing approaches to assess the efficacy, as shown in Table 8. Our hybrid system outperformed the comparable speech COVID-19 identification models, which attained an accuracy of 98.2% [9]. Moreover, Ayalew et al. [44], Ali et al. [45], and Salama et al. [46] utilized X-rays-based approaches. However, while comparing our model with these approaches, the most effective results are attained by our hybrid technique. Furthermore, the proposed model taking both types of inputs i.e., X-rays and audios, and this property makes our model more robust. Gunavant et al. [17] also utilized audio data and extracted MFCC features, however, achieved 78.3%. On the other side, Laguarta Jord et al. [16] also performed cough analysis achieving 97.3% accuracy. Whereas, the proposed model extracts the most important features from the chest images and audio mel-spectrograms due to its unique layered architecture providing maximum performance. The reason behind the better performance is the well-known architecture of ResNet50, and the unique model i.e., VGGish for the audio classification tasks. Both proposed networks effectively learn the hidden patterns and also overcome the issue of overfitting. On the other side, the performance of the proposed model depends upon the quality of scans and audios.

5 Discussion

In this study, we have proposed two models for the detection of COVID-19 virus. The first model is based on the speech data of patients. The speech data comprised audio of breathing and coughing. Second, the model is based on ResNet50 using chest X-rays. To choose an image-based classifier, we employed several DL methods, such as Alex-Net, InceptionNetV3, SqueezeNet, DenseNet, DarkNet, VGG16, ResNet50, and GoogleNet. The maximum accuracy was attained by the ResNet50-based technique. The proposed model performed exceptionally on chest X-rays due to its architecture. Then second, the speech-based model, was based on VGGish, which takes input in the form of audio mel-spectrograms. Moreover, we combined the detection powers of both proposed models as described in Table 7, attaining 99.7% accuracy.

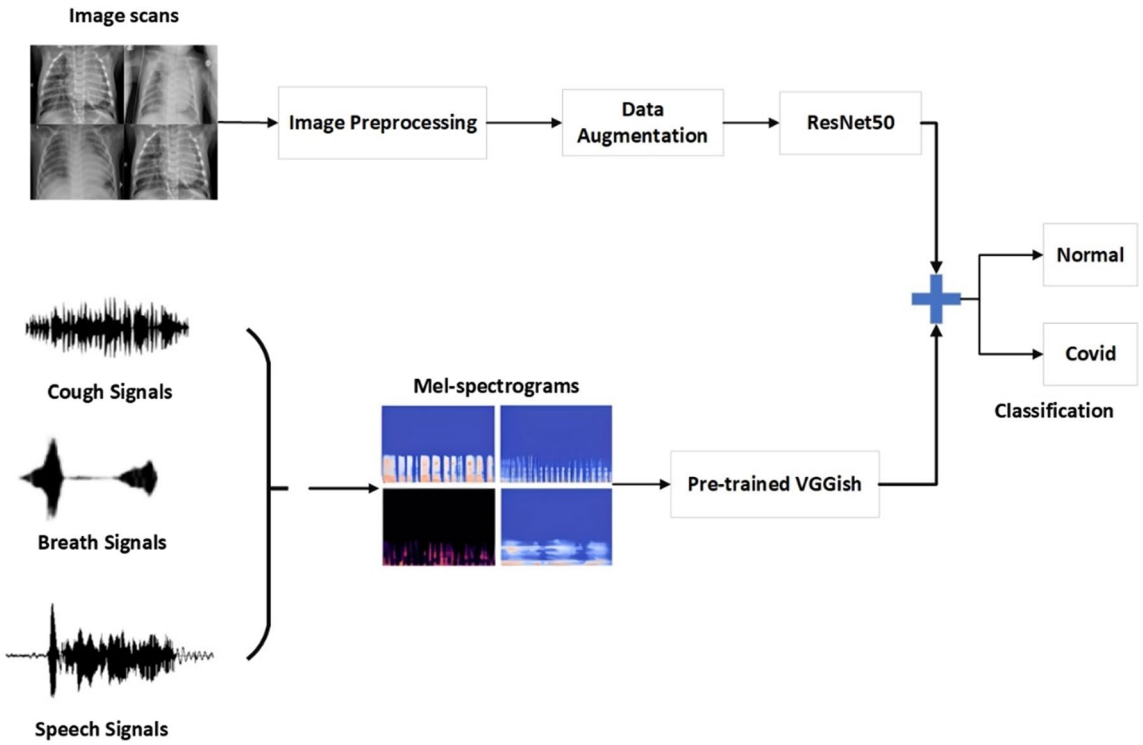

Fig. 6 The hybrid model's architecture

Table 7 The decision-making strategy for a hybrid model

Image-based system	Speech-based system	Hybrid system
COVID-19-positive	COVID-19-positive	COVID-19-positive
COVID-19-positive	COVID-19-negative	COVID-19-positive
COVID-19-negative	COVID-19-positive	COVID-19-positive
COVID-19-negative	COVID-19-negative	COVID-19-negative

Although the proposed model outperformed the existing models, some limitations and challenges exist that should be considered in future work. The dataset is sourced from a single medical center, which might not fully represent the diversity of cases seen in different regions or healthcare settings. This could limit the external validity of the study. Moreover, we also analyzed that the quality of scans greatly affected the performance of the proposed model when we used some blurry samples. On the other hand, the unavailability of the speech dataset of COVID-19 patients is also a great concern, as it is necessary to cross-validate the performance of the VGGish over real-world data.

6 Conclusion

Our research introduced two distinctive approaches for COVID-19 detection. First, we proposed an innovative audio-based method leveraging the VGGish network, which utilizes the features of coughing, breathing, and vocal patterns to identify COVID-19 infection. In parallel, we developed an image-based approach, utilizing chest X-rays for identification based on ResNet50. Both methodologies aim to detect COVID-19 in its early stages, thereby enhancing the chances of timely intervention and treatment. To optimize our models, we conducted grid search to fine-tune the hyper-parameters of both the speech-based and image-based architectures. The results demonstrated the impressive efficiency of the VGGish model in identifying COVID-19. Our audio-based model exhibited remarkable robustness, achieving a remarkable 98.9% accuracy across audio spectrograms. To further enhance our system's accuracy, we employed data augmentation techniques. Moreover, when we combined both approaches to propose a hybrid system, an accuracy of 99.7% was attained.

However, a notable challenge we encountered was the inherent data imbalance between speech and image datasets. In our future endeavors, we aspire to gather real-world datasets and train our model without augmentation, further elevating the performance of our proposed system. We anticipate that continued efforts, including the acquisition of more diverse datasets and refined model training, will lead to even more robust and effective tools for early COVID-19 detection.

Table 8 Existing methods and performance

Works	Techniques	Descriptions	Accuracy
Alsabek et al. [2]	MFCC	COVID-19 identification is performed by analyzing MFCC features and providing correlation coefficient evaluation	The average linear relationship is 0.42
Hassan Abdelfatah et al. [9]	LSTM	COVID-19 is diagnosed early, and many acoustic aspects are evaluated	Accuracy: 98.2%
Pahar Madhurananda et al. [12]	RSNET, LR, LSTM, SVM, CNN and RNN	Identifying positive and negative coughing results	Accuracy: 95.34%
Deshpande et al. [13]	BiLSTM	Giving an early screening for COVID-19 cough-based examination	AUC: 64.43%
Kumar et al. [15]	Random forest, Logistic regression, and multilayer perceptron	Cough analysis is being used to provide early testing for COVID-19	AUC: 81.89
Laguarta Jord et al. [16]	CNN, Rsnet50, MFCC	Cough analysis is used to provide an advanced screening for COVID-19	Accuracy: 97.3%
Gunavant et al. [17]	MFCC	Cough analysis will be used to provide an early preview for COVID-19	ROC: 77.10% AUC: 77.1% Accuracy: 78.3%
Maghdid et al.[18]	Alex-Net, CNN	COVID-19 identification using X-rays and CT pictures of patients	Accuracy: 98%
Jaiswal et al. [20]	Densenet-201 CNN model	Utilizing chest CT outputs to recognize the presence of COVID-19	Accuracy: 97%
Ayalew et al. [44]	CNN and SVM	They extracted features from X-rays using CNN and passed them to SVM for the detection of COVID-19	Accuracy: 99.1%
Ali et al. [45]	CNN and K-Nearest Neighbor	They applied morphological approaches to compute lesions and then applied classification	Accuracy: 95.65%
Salama et al. [46]	CNN and SVM	Several CNN layers were used for feature extraction, and numerous ML algorithms were utilized	Accuracy: 99.39%
Our proposed model	ResNet50 and VGGish	Used chest CT images and Speech data	Accuracy: 99.1%

Author Contributions This work was carried out in collaboration among all authors. FA, RM, FSB, AER, and HH conceived the main idea and contributions for this study and supervised the work. Methodology, MEM, RM, FSB, and FA; Validation, AER, HH, and MEM; Writing—review and editing, FA, RM, AER, FSB, and HH; Writing—first draft preparation, RM and FSB All authors read and approved the final version of the manuscript.

Funding The researchers extend their appreciation to King Saud University, Saudi Arabia, for funding this work through "Researchers Supporting Project number (RSPD2024R711), King Saud University, Riyadh, Saudi Arabia".

Data Availability Data used during the experiments can be shared on demand.

Declarations

Conflict of Interest The writers have no irreconcilable situations to announce that are pertinent to the substance of this article.

Ethical Approval Not applicable.

References

1. Dua, M., Jain, C., Kumar, S.: LSTM and CNN based ensemble approach for spoof detection task in automatic speaker verification systems. J. Ambient. Intell. Humaniz. Comput.Intell. Humaniz. Comput. **13**(4), 1985–2000 (2022)

2. Alsabek, M.B., Shahin, I., Hassan, A.: Studying the similarity of COVID-19 sounds based on correlation analysis of MFCC. At the 2020 international conference on Communications, computing, cybersecurity, and Informatics (SCI). IEEE (2020)
3. Aggarwal, S., et al.: Automated COVID-19 detection in chest X-ray images using fine-tuned deep learning architectures. Expert. Syst. **39**(3), e12749 (2022)
4. Chan, J.F.-W., et al.: A familial cluster of pneumonia associated with the 2019 novel coronavirus indicating person-to-person transmission: a study of a family cluster. Lancet **395**(10223), 514–523 (2020)
5. Mahum, R., et al.: A novel hybrid approach based on deep CNN to detect glaucoma using fundus imaging. Electronics **11**(1), 26 (2021)
6. Mahum, R., Aladhadh, S.: Skin lesion detection using hand-crafted and DL-based features fusion and LSTM. Diagnostics **12**(12), 2974 (2022)
7. Akhtar, M.J., et al.: A robust framework for object detection in a traffic surveillance system. Electronics **11**(21), 3425 (2022)
8. Mahum, R., et al.: A novel hybrid approach based on deep CNN features to detect knee osteoarthritis. Sensors **21**(18), 6189 (2021)
9. Hassan, A., Shahin, I., Alsabek, M.B.: COVID-19 detection system using recurrent neural networks. In: 2020, there was an International conference on communications, computing, cybersecurity, and informatics (CI). IEEE (2020)
10. Nassif, A.B., et al.: COVID-19 detection systems using deep-learning algorithms based on speech and image data. Mathematics **10**(4), 564 (2022)
11. Alafif, T., et al.: Machine and deep learning towards COVID-19 diagnosis and treatment: survey, challenges, and future directions. Int. J. Environ. Res. Public Health **18**(3), 1117 (2021)
12. Pahar, M., et al.: COVID-19 cough classification using machine learning and global smartphone recordings. Comput. Biol. Med.. Biol. Med. **135**, 104572 (2021)
13. Deshpande, G., Schuller, B.W.: The DiCOVA 2021 challenge–an encoder-decoder approach for COVID-19 recognition from coughing audio. In: Proc. Interspeech (2021)
14. Munir, M.H., et al.: An automated framework for corona virus severity detection using combination of AlexNet and faster RCNN (2022)
15. Das, R.K., Madhavi, M., Li, H.: Diagnosis of COVID-19 using auditory acoustic cues. In: 22nd Annual Conference of the International Speech Communication Association, INTERSPEECH 2021 (2021)
16. Laguarta, J., Hueto, F., Subirana, B.: COVID-19 artificial intelligence diagnosis using only cough recordings. IEEE Open J. Eng. Med. Biol. **1**, 275–281 (2020)
17. Chaudhari, G., et al.: Virufy: Global applicability of crowd sourced and clinical datasets for AI detection of COVID-19 from cough. arXiv preprint https://arXiv.org/2011.13320, (2020)
18. Maghdid, H.S., et al.: Diagnosing COVID-19 pneumonia from X-ray and CT images using deep learning and transfer learning algorithms. In: Multimodal image exploitation and learning 2021. SPIE (2021)
19. Wang, S., et al.: A deep learning algorithm using CT images to screen for corona virus disease (COVID-19). Eur. Radiol.Radiol. **31**(8), 6096–6104 (2021)
20. Jaiswal, A., et al.: classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning. J. Biomol. Struct. Dyn.Biomol. Struct. Dyn. **39**(15), 5682–5689 (2021)
21. Narin, A., Kaya, C., Pamuk, Z.: Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. Pattern Anal. Appl. **24**(3), 1207–1220 (2021)
22. Hershey, S., et al.: CNN architectures for large-scale audio classification. In 2017, I attended an international conference on acoustics, speech, and signal processing (Picasso). IEEE (2017)
23. Kanwal, T., Mahum, R., AlSalman, A.M., Sharaf, M., Hassan, H.: Fake speech detection using VGGish with attention block. EURASIP J Audio, Speech, Music Process **1**, 35 (2024)
24. Mastromichalakis, S.: *ALReLU:* a different approach to the Leaky ReLU activation function to improve Neural Networks Performance. arXiv preprint https://arXiv.org/2012.07564, (2020)
25. MONEY, P.: Chest X-ray images. (2018); https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia
26. Taylor, L., Nitschke, G.: Improving deep learning with generic data augmentation. In: 2018 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE (2018)
27. Gómez-Ríos, A., et al.: Towards highly accurate coral texture images classification using deep convolutional neural networks and data augmentation. Expert Syst. Appl. **118**, 315–328 (2019)
28. O'Shea, K., Nash, R.: An introduction to convolutional neural networks. arXiv preprint https://arXiv.org/1511.08458, (2015)
29. Aszemi, N.M., Dominic, P.: Hyperparameter optimization in a convolutional neural network using genetic algorithms. Int. J. Adv. Comput. Sci. Appl. **10**(6) (2019)
30. Sharma, N., et al.: Coswara—a database of breathing, cough, and voice sounds for COVID-19 diagnosis. arXiv preprint https://arXiv.org/2005.10548, (2020)
31. Albawi, S., Mohammed, T.A., Al-Zawi, S.: Understanding of a convolutional neural network. In: 2017, the International Conference on Engineering and Technology (ICET) was held. IEEE (2017)
32. Huang, G., et al.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017)
33. Szegedy, C., et al.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016)
34. Mahum, R., Irtaza, A., Javed, A.: EDL-Det: a robust TTS synthesis detector using VGG19-based YAMNet and ensemble learning block. IEEE Access **11**, 134701–134716 (2023)
35. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016)
36. Xie, M., et al.: Transfer learning from deep features for remote sensing and poverty mapping. In: Thirtieth AAAI Conference on Artificial Intelligence. (2016)
37. Yue, J., et al.: Spectral–spatial classification of hyperspectral images using deep convolutional neural networks. Remote Sens. Lett. **6**(6), 468–477 (2015)
38. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng.Knowl. Data Eng. **22**(10), 1345–1359 (2009)
39. Zhuang, F., et al.: A comprehensive survey on transfer learning. Proc. IEEE **109**(1), 43–76 (2020)
40. Liashchynskyi, P., Liashchynskyi, P.: Grid search, random search, genetic algorithm: a big comparison for NAS. arXiv preprint https://arXiv.org/1912.06059, (2019)
41. Berrar, D.: Cross-validation. (2019)
42. Peña Yañez, A.: El anillo esofágico inferior. Rev. Esp. Enferm. Apar. Dig.Enferm. Apar. Dig. **26**, 505–516 (1967)
43. Kaariainen, M.: Semi-supervised model selection based on cross-validation. In: The 2006 IEEE International Joint Conference on Neural Network Proceedings. IEEE (2006)
44. Ayalew, A.M., Salau, A.O., Tamyalew, Y., et al.: X-ray image-based COVID-19 detection using deep learning. Multimed. Tools Appl. **82**, 44507–44525 (2023). https://doi.org/10.1007/s11042-023-15389-8
45. Ali, A.M., Ghafoor, K., Mulahuwaish, A., et al.: COVID-19 pneumonia level detection using deep learning algorithm and transfer learning. Evol. Intel.. Intel. **17**, 1035–1046 (2024). https://doi.org/10.1007/s12065-022-00777-0
46. Salama, G.M., Mohamed, A., Abd-Ellah, M.K.: COVID-19 classification based on a deep learning and machine learning fusion technique using chest CT images. Neural Comput. Appl.Comput. Appl. **36**, 5347–5365 (2024). https://doi.org/10.1007/s00521-023-09346-7