IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 10, 2022, accepted June 8, 2022, date of publication June 13, 2022, date of current version June 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3182498

Pneumonia Detection Proposing a Hybrid Deep Convolutional Neural Network Based on Two Parallel Visual Geometry Group Architectures and Machine Learning Classifiers

MOHAMMAD YASELIANI[1], ALI ZEINAL HAMADANI[1], ABTIN IJADI MAGHSOODI[2], AND AMIR MOSAVI[3,4,5]

[1]Department of Industrial Engineering, Isfahan University of Technology, Isfahan 8415683111, Iran
[2]Department of Information Systems and Operations Management, Faculty of Business and Economics, Business School, University of Auckland, Auckland 1010, New Zealand
[3]Institute of Software Design and Development, Óbuda University, 1034 Budapest, Hungary
[4]Institute of Information Engineering, Automation and Mathematics, Slovak University of Technology in Bratislava, 811 07 Bratislava, Slovakia
[5]Faculty of Civil Engineering, Technische Universität Dresden, 01067 Dresden, Germany

Corresponding authors: Abtin Ijadi Maghsoodi (abtin.ijadi-maghsoodi@auckland.ac.nz) and Amir Mosavi (amir.mosavi@kvk.uni-obuda.hu)

This research is supported by the European Union's Horizon 2020 Research and Innovation Programme under the Programme SASPRO 2 COFUND Marie Sklodowska-Curie under Grant 945478. The authors would like to acknowledge the contribution of the Slovak Research and Development Agency under the project APVV-20-0261.

ABSTRACT Pneumonia is an acute respiratory infection that has led to significant deaths of people worldwide. This lung disease is more common in people older than 65 and children under five years old. Although the treatment of pneumonia can be challenging, it can be prevented by early diagnosis using Computer-Aided Diagnosis (CAD) systems. Chest X-Rays (CXRs) are currently the primary imaging tool for detection of pneumonia, which are widely used by radiologists. While the standard approach of detecting pneumonia is based on clinicians' decisions, various Deep Learning (DL) methods have been developed for detection of pneumonia considering CAD system. In this regard, a novel hybrid Convolutional Neural Network (CNN) model is proposed using three classification approaches. In the first classification approach, Fully-Connected (FC) layers are utilized for the classification of CXR images. This model is trained for several epochs and the weights that result in the highest classification accuracy are saved. In the second classification approach, the trained optimized weights are utilized to extract the most representative CXR image features and Machine Learning (ML) classifiers are employed to classify the images. In the third classification approach, an ensemble of the proposed classifiers is created to classify CXR images. The results suggest that the proposed ensemble classifier using Support Vector Machine (SVM) with Radial Basis Function (RBF) and Logistic Regression (LR) classifiers has the best performance with 98.55% accuracy. Ultimately, this model is deployed to create a web-based CAD system to assist radiologists in pneumonia detection with a significant accuracy.

INDEX TERMS Pneumonia, chest X-ray, deep learning, convolutional neural network, visual geometry group, machine learning.

I. INTRODUCTION

Pneumonia is a lung disease that is caused by acute respiratory infection. Pneumonia causes reduced oxygen intake and painful breathing [1]. Although pneumonia can affect people at any age, it is more common in people older than 65 and children under five years old [2]. According to World Health Organization (WHO), pneumonia is responsible for 14% of all deaths in children under five years old, causing 740,180 deaths in 2019 [1]. Pneumonia is also estimated to kill about 11 million children by 2030 [3]. Further statistics indicate that almost one million pneumonia-infected people older than 65 are hospitalized in the United States every year [4]. This disease can affect people everywhere globally, but it is most common in South Asia and Sub-Saharan Africa [1]. According to WHO, exposure to air pollution has led to 45% and 28% of all pneumonia deaths in adults and children, respectively [5]. Moreover, bacteria, viruses, and fungi are the three most

The associate editor coordinating the review of this manuscript and approving it for publication was Nikhil Padhi.

common causes of pneumonia [1]. As a noteworthy example, the widespread Coronavirus Disease known as COVID-19 or SARS-CoV-2 is regarded as a common cause of viral pneumonia [6]. There are some variables to differentiate viral and bacterial pneumonia, including but not limited to age, history of disease, and response to antibiotic treatments. Viral pneumonia often occurs in children under five years old and has a slow onset; while bacterial pneumonia is more common in adults and has a faster onset [7]. Although the treatment of pneumonia can be challenging, it can be prevented with modest treatments and cured with low-cost, low-tech medication and care [1]. Therefore, there is an urgent need to develop diagnostic tools in order to reduce pneumonia-related mortality, particularly in children and old people.

Some popular tools to evaluate the presence of pneumonia in a person are Chest X-Ray (CXR), Computed Tomography (CT) of the lungs, ultrasound of the chest, Magnetic Resonance Imaging (MRI) of the chest, and needle biopsy of the lung [8]. Medical X-rays are electromagnetic radiations that have higher energy than visible light and can penetrate through most objects. Compared to CT images, X-rays are less expensive, take less time to obtain, and cause lower radiation exposure [9]. Furthermore, in comparison to MRI, X-rays are much less expensive, more available and quickly accessed by radiologists [10]. Currently, CXRs are considered the best standard tool for detecting pneumonia, which can distinguish pneumonia from other lung infections and diseases [11]. The use of Machine Learning (ML) and Deep Learning (DL) is growing significantly. ML and DL methods have shown outstanding performance in healthcare, especially medical image analysis [12]. ML- and DL-based methods can be used to create Computer-Aided Diagnosis (CAD) systems to assist physicians and radiologists in medical decision-making [13]. It has been shown that CAD systems can perform on a par with or better than radiologists in terms of sensitivity and specificity [14]. Additionally, CAD-assisted radiologists have indicated a higher classification accuracy, compared to unassisted radiologists in interpretation of CXR images [15]. Radiologists identify the image parameters to make a clinical decision, while ML- and DL-based CAD systems can also assess the importance of the image parameters to provide a clinical decision. Moreover, radiologists rely on their prior definition of discriminative features to classify the images, while CAD systems perform feature discrimination without any prior definition by experts and radiologists. In some cases, a disease may share common features with other diseases, making the process of diagnosis challenging to radiologists. CAD systems can solve this issue by identifying the features of disease in an efficient way [16]. It is also notable that CAD systems can be much faster than an experienced radiologist in analyzing medical images [14].

In the past recent years, Convolutional Neural Networks (CNNs) have shown a great potential in image classification and segmentation and are widely utilized for creating DL-based CAD systems. While some studies have analyzed the binary classification of normal and pneumonia images, other research studies have also studied the multi-class classification of normal images and different types of pneumonia, including viral and bacterial pneumonia infections. Fernandes *et al.* [17] proposed a Bayesian CNN-based method, based on Visual Geometry Group 16 (VGG16) to detect pneumonia. Dey *et al.* [18] used a customized VGG19 architecture and five classifiers, including linear Support Vector Machine (SVM), SVM-Radial Basis Function (RBF), K-Nearest Neighbors (KNN), Random Forest (RF), and Decision Tree (DT) to detect pneumonia. Rajaraman *et al.* [19] proposed a novel method for locating the Region of Interest (ROI), using customized models of VGG16, sequential CNN, Residual CNN, and Inception CNN for detection of pneumonia. Jain *et al.* [20] used six models, including two custom-designed models, VGG16, VGG19, ResNet50, and InceptionV3 to detect pneumonia cases. There have also been other research studies that have proposed custom-designed CNNs for detection of pneumonia [21]–[23]. Kermany *et al.* [24] proposed an Artificial Intelligence (AI) system based on a transfer learning framework for detecting pneumonia cases. Rahman *et al.* [25] used four CNN models, including AlexNet, ResNet, DenseNet, and SqueezeNet with transfer learning for pneumonia detection. Zhang *et al.* [26] proposed a straightforward VGG16-based model architecture with fewer layers for pneumonia detection.

Although various studies have analyzed CNN frameworks for pneumonia detection, to the best of authors' knowledge, not a single study has considered using a hybrid VGG-based CNN model for detection of pneumonia. As hybrid methods can better handle incomplete data [27] and improve the model's computation, robustness, functionality, and accuracy [28], using a hybrid CNN model for detection of pneumonia is of a significant importance. Moreover, none of the previous studies have implemented a CAD system for a reliable and accurate detection of pneumonia. In order to address these research gaps, the current study has proposed a hybrid CNN model, combining two popular CNN models. For this purpose, the features obtained using VGG16 and VGG19 networks are concatenated to create a new hybrid VGG-based CNN model. Deep feature concatenation is an effective way of improving classification process in CNN models and provides multiscale information of input images [29], [30]. To classify CXR images, three approaches are utilized. In the first approach, we have designed the proposed hybrid CNN model with Fully Connected (FC) layers and trained the hybrid model for a defined number of epochs and updated the weights of the model using backpropagation process and saved the best-performing weights. In the second approach, we have loaded and utilized the saved optimized weights to extract CXR image features and trained five ML classifiers for classifying CXR images. To the best of authors' knowledge, none of the previous studies has deployed such weight utilizing approach for classifying the images. In the third approach, the trained classifiers obtained in the first and second classification approaches are employed to create an ensemble classifier, which takes the class probabilities

attained by those classifiers and assigns a weight to each of them in such a way that the highest classification accuracy is produced. Ultimately, in this study, a CAD system based on the most accurate trained model is designed to detect pneumonia cases.

The remainder of this research study is structured as follows: Section II presents a brief overview of the fundamentals of the research methodology including CNN, ML models and the proposed hybrid CNN model. Results and findings of computations and calculations using the proposed hybrid CNN model are presented in section III. Finally, the concluding clarifications and guidelines for future research are presented in Section IV.

II. METHODS AND MATERIALS FUNDAMENTALS

A. CONVOLUTIOANL NEURAL NETWORK (CNN)

Neural networks (NNs) are a type of non-linear statistical data modeling tool based on densely connected nodes that can simulate complicated input-output interactions [31]. Feed-forward NNs (FNNs) are a type of NNs in which the decision flow is unidirectional, flowing from the input to the output in loop-free consecutive layers [32]. Accordingly, CNNs are considered as a type of FNNs [33]. CNNs are one of the most used DL methods that have demonstrated excellent results on the ImageNet Large Scale Visual Recognition Competition 2012 (ILSVRC2012). The application area of CNNs is significant e.g., classification, segmentation, and Natural Language Processing (NLP) [34]. CNNs have widely been utilized in medical imaging, showing promising results on medical image classification and segmentation [35], [36]. The architectures of CNNs are based on various building blocks, including convolution layers, pooling layers, and Fully-Connected (FC) layers. The role of convolution layers is feature extraction and usually consist of a mixture of linear and nonlinear operations, i.e., convolution operations and activation functions. The parameters of convolution layers are kernels and the hyperparameters of convolution layers which includes kernel size, number of kernels, stride, padding, and activation function. The formula of convolution operations is presented in (1) [37].

$$O^l_{p,q,r} = f(W^l_r I^{l-1}_{p,q} + b^l_r) \tag{1}$$

where $O^l_{p,q,r}$ represents the output feature map of location (p, q) for rth kernel in layer l, W^l_r is the values of weight vector of rth kernel in layer l, $I^{l-1}_{p,q}$ denotes the input vector of location (p, q) in layer $l-1$, and b^l_r is representative of bias for rth kernel in layer l. Moreover, $f(\cdot)$ is the activation function [37]. Pooling layers perform down-sampling operation and reduce dimensionality of feature maps. While pooling layers do not include any parameters, they consist of some hyperparameters, including stride, padding, and filter size. Max pooling and global average pooling are two popular types of pooling layers. In Max pooling, patches are extracted from the feature maps and the maximum value in each patch is selected as the output. In global average pooling, a feature map is converted into a 1×1 array by calculating the average of all elements in the feature map [34]. The output dimension after conducting a convolution/pooling operation in a CNN is calculated using (2) [38].

$$o = \left\lfloor \frac{n - f + 2p}{s} \right\rfloor + 1 \tag{2}$$

where n denotes the input dimension, f is the kernel/filter size, p indicates the padding size, and s is representative of stride size.

The feature maps of the final convolution or pooling layers are flattened and passed to a number of one-dimensional (1D) vectors, called FC layers or dense layers. The final FC layer has a number of nodes, equal to the number of classes in a specific classification task. The parameters of FC parameters are weights and the hyperparameters of FC layers include the number of weights and activation function. Rectified Linear Unit (ReLU) is the most used activation function for FC layers, which is presented in (3) [34].

$$f(x) = max(x, 0) \tag{3}$$

It should be noted that the activation function of the last FC layer is typically sigmoid for binary classification and SoftMax for multi-class classification [34]. The node values in the last FC layer in a CNN can be calculated using (4), and the sigmoid activation function for a binary classification problem is demonstrated in (5) [39].

$$z = w^T h + b \tag{4}$$

$$P(y = 1|x) = \frac{1}{1 + exp(-z)} \tag{5}$$

where h denotes the internal calculations of layers in NN, b indicates the bias, and w is representative of weights for calculating the value of an output node. In addition, y and x represent the output class and input vector, respectively. The SoftMax activation function for a multi-class classification problem is presented in (6) [39].

$$P(y|x) = \frac{exp(f_y)}{\sum_{c=1}^{C} exp(f_c)} \tag{6}$$

where y denotes the class in a multi-class classification problem and x represents the input vector. Moreover, f_c shows the cth element of the vector of class scores in the last FC layer. In SoftMax activation function, the class k with the biggest P value is selected as the output class [34]. Figure 1 illustrates a comprehensive graphical overview of a CNN architecture for feature extraction and classification purposes. In the training process of a CNN, the weights of convolution layers and FC layers are updated using backpropagation algorithm. The loss function and Gradient Descent (GD) are the two fundamental components of backpropagation, where the loss function is minimized using GD. The Cross-Entropy (CE) loss function is one of the most used loss functions in CNNs. Accordingly, the CE loss function for a binary classification problem with

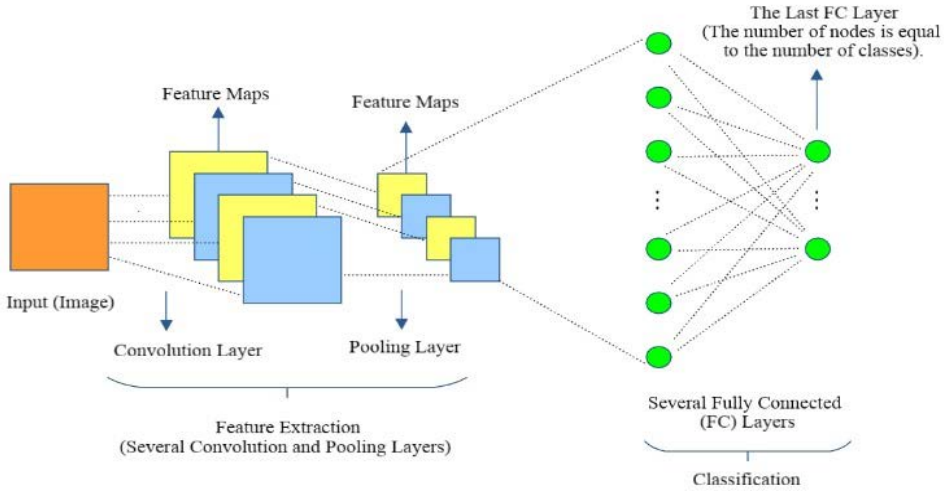


FIGURE 1. An overview of a CNN architecture.

sigmoid activation function can be computed as demonstrated in (7) [39].

$$L = \frac{1}{N}\sum_{i=1}^{N} -\left[y_i\, log\left(\frac{1}{1+exp\,(-z)}\right) + (1-y_i)\, log\left(\frac{exp\,(-z)}{1+exp\,(-z)}\right)\right] \quad (7)$$

where z is calculated according to (4). For a multi-class classification problem with SoftMax activation function, the CE loss function can be formulated as (8) [39].

$$L = \frac{1}{N}\sum_{i=1}^{N} -log\left(\frac{exp\,(f_{y_i})}{\sum_{c=1}^{C} exp\,(f_c)}\right) \quad (8)$$

where N is the number of training points, y_i denotes the class of ith input image, and f_c shows the cth element of the vector of class scores in the last FC layer [39]. It is notable that SoftMax can also be deployed as an activation function for a binary classification problem.

Gradient Descent (GD) is a way of minimizing an objective function $J(\theta)$ which is parametrized by θ. GD has three variants, including batch GD, Stochastic GD (SGD), and mini-batch GD. In batch GD, each parameter is updated for all the training samples in the dataset using (9). In contrast to batch GD, SGD conducts parameter update for each training sample in the dataset. In mini-batch GD, parameter update is performed for a mini-batch of n samples in the dataset. The formulas of SGD and mini-batch GD are presented in (10) and (11), respectively. In these equations, θ denotes the learnable parameter, $J(\theta)$ represents the objective function that has to be minimized, η is the learning rate, and $\nabla_\theta J(\theta)$ shows the gradient of the objective function with respect to parameters θ [40].

$$\theta := \theta - \eta.\nabla_\theta J(\theta) \quad (9)$$

$$\theta := \theta - \eta.\nabla_\theta J\left(\theta; x^{(i)}; y^{(i)}\right) \quad (10)$$

$$\theta := \theta - \eta.\nabla_\theta J\left(\theta; x^{(i:i+n)}; y^{(i:i+n)}\right) \quad (11)$$

FIGURE 2. The architecture of VGG16 network.

SGD conducts frequent, high-variance updates, causing the objective function to fluctuate significantly. The variability of SGD allows it to reach new and possibly better local minima. It is also notable that mini-batch GD is most often called as SGD [40]. There are also other optimizers for updating the parameters of NNs or CNNs, such as adaptive optimizers, but it has been shown that SGD usually outperforms adaptive optimizers in terms of generalization on the test set [41].

1) VISUAL GEOMETRY GROUP (VGG) ARCHITECTURES

Visual Geometry Group (VGG) is a classical CNN architecture that uses 3×3 convolution kernels to extract features. VGG16 and VGG19 are two types of VGG architectures that include 13 and 16 convolution layers, respectively. There are also five 2×2 pooling layers and three FC layers in both VGG16 and VGG19 architectures. Therefore, there are a total 16 and 19 layers in VGG16 and VGG19 networks without considering pooling layers, respectively. These layers are arranged in five blocks, each of which start from a convolution layer to a max pooling layer. The first two FC layers of these two networks contain 4096 nodes and the last one consists of 1000 nodes, which represents the number of classes in the ILSVRC2014. The last FC layer of these two networks can be changed according to the number of available classes in a specific classification problem. The last layers of feature extraction in VGG16 and VGG19 networks are the 13th and 16th convolution layers, respectively. These two networks extract the features with a dimension of $7 \times 7 \times 512$, where the first two numbers (7) represent the width and height, respectively, and the third number (512) denotes the depth of the output feature map. The input

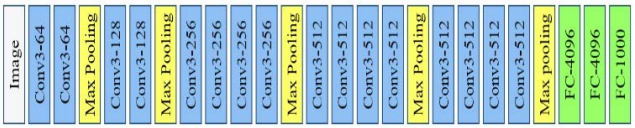


FIGURE 3. The architecture of VGG19 network.

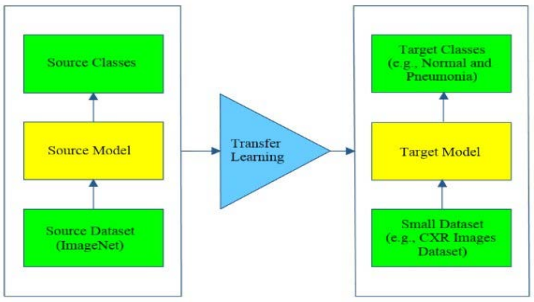


FIGURE 4. The idea of transfer learning.

image size for these networks is also 224 × 224 pixels [42]. Figures 2 and 3 portray the architectures of VGG16 and VGG19 networks, respectively.

2) TRANSFER LEARNING

In transfer learning, a network that has been trained for one task is used as the starting point for solving another problem. Accordingly, rather than going through the lengthy process of training a network with randomly initialized weights, pre-trained networks are utilized as the starting point for some specific classification tasks in transfer learning. The core premise of the transfer learning is that the filters generated by convolution layers of CNNs can be helpful for a wide range of image identification tasks, not only the ones for which they were initially trained [43]. ImageNet [44] is a database of more than 14 million images from 21841 categories that has been used to train different CNN architectures. As an example, there are pre-trained VGG16 and VGG19 architectures that have been trained on the ImageNet which can be used in different classification tasks. A graphical representation of transfer learning has been conceptualized in Figure 4.

Pan and Yang [45] proposed a transfer learning framework based on domain, task, and marginal probability. D is a domain consisting of feature space χ and a marginal probability $P(X)$, where X denotes a dataset of n samples $(X = \{x_1, x_2, \ldots, x_n\} \in \chi)$. A domain D can be defined as (12), where χ denotes the space of all vectors and X indicates a learning example with n samples. For a given domain, a task T is defined as (13).

$$D = \{\chi, P(X)\} \tag{12}$$

$$T = \{\gamma, P(Y \mid X)\} = \{\gamma, \phi\}, \quad Y = \{y_1, \ldots, y_n\}, \quad y_i \in \gamma \tag{13}$$

where γ shows the label space and ϕ is representative of a predictive function that has been trained using feature vector/label pairs (x_i, y_i), where $x_i \in \chi$ and $y_i \in \gamma$. It should be noted that $\phi(x_i) = y_i$, where ϕ makes label predictions for each feature vector. The purpose of transfer learning is to improve the learning of a predictive function ϕ_T for a given target domain D_T and learning task T_T, using a source domain D_S and learning task T_S, where $D_S \neq D_T$ and $T_S \neq T_T$.

3) OVERFITTING IN COLVOLUTIONAL NEURAL NETWORKS

Overfitting refers to a state that there is a large gap between the training loss and test loss. In overfitting, the network learns feature maps that perform well on the training set, while perform poorly on the test set. Therefore, a much higher test loss will lead to training loss in the presence of overfitting during the training process [46]. Although overfitting is not completely avoidable, there are some ways to solve this problem in CNNs. Dropout regularization, early stopping, and image augmentation are three common ways that are used to tackle this problem during training the networks. Early stopping is a regularization strategy that terminates training, when updates of the weights do not provide further improvements on the test set. In this regard, throughout the training, the current best weights are saved and when updates of the weights no longer generate an improvement after a certain number of iterations, training process will be stopped and the previous best values will be used [47]. Dropout regularization was introduced in 2014. It refers to removing a unit from the network, together with all of its incoming and outgoing connections. In the most basic case, each unit is kept with a fixed probability p. The p value is usually set at 0.5, which has shown to be optimal for a wide range of networks [48]. Image augmentation is also another technique to prevent overfitting by increasing the number of images in the training set using some operations, including but not limited to shearing, zooming, rotation, and flipping [49].

B. MACHINE LEARNING (ML) CLASSIFIERS

1) K-NEAREST NEIGHBORS (KNN)

K-Nearest Neighbors (KNN) is an ML classification algorithm that performs the classification task by calculating the distance between a given sample and all the training data. Accordingly, the given sample is assigned with a label based on the majority voting on the labels of the K selected nearest neighbors. Considering a training set D and a test object $z = (x', y')$ with x' as the data point and y' as its associated class, KNN algorithm measures the distance between z and all the points $(x, y) \in D$, where x is the data belonging to the training set and y denotes its corresponding class. After obtaining the K nearest neighbors list Dz, the test object is assigned with the majority class of its nearest neighbors according to (14) [50].

$$y' = \arg\max_{v} \sum_{(x,y)\in D_z} I(v = y_i) \tag{14}$$

where v denotes a class, y_i indicates the class of ith nearest neighbor, and $I(\cdot)$ represents an indicator function that is equal to 1, when its expression is true and otherwise, it is 0.

Setting an optimal K has a significant impact on the quality of results. For example, a very small K value makes the result sensitive to noise points and if it is too large, the list

of nearest neighbors contains many data points from other classes. The distance measurement is another critical issue in developing a KNN algorithm. Euclidean distance is one of the distance measurement functions for constructing a KNN classifier [50]. The Euclidean distance function is demonstrated in (15) [51]. $m = (m_1, m_2, \ldots, m_p)$ and $n = (n_1, n_2, \ldots, n_p)$ represent two data points with p features.

$$d(m, n) = \sqrt{(m_1 - n_1)^2 + \ldots + (m_p - n_p)^2} \quad (15)$$

2) LOGISTIC REGRESSION (LR)

Logistic Regression (LR) is an ML classification algorithm that predicts a binary output variable based on some explanatory variables. LR can also be generalized to multi-class classification tasks. The binary LR algorithm is presented in (16) [52], where P is the probability of the output variable/class 1, given an input vector $x = (x_1, x_2, \ldots, x_n)$ with n features.

$$\begin{aligned} P(Y = 1 \mid X = x) &= \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n)}} \\ &= \frac{1}{1 + e^{-(\beta_0 + \sum \beta_i x_i)}} \end{aligned} \quad (16)$$

3) NAÏVE BAYES (NB)

Naïve Bayes (NB) is an ML classification algorithm that uses feature vectors and class prior probabilities to predict the output class as presented in (17) [53].

$$P(C = i \mid X = x) = \frac{P(X = x \mid C = i) * P(C = i)}{P(X = x)} \quad (17)$$

where C is representative of the output class and $x = (x_1, x_2, \ldots, x_n)$, which denotes the set of all features taken form a domain $\Omega = D_1 * \ldots * D_n$. To simply the calculation of $P(X = x|C = i)$ in NB classifier, the features are assumed to be independent in the dataset and hence, the probability is calculated as demonstrated in (18) [53].

$$P(X = x|C = i) = \prod_{j=1}^{n} P\left(x_j|C = i\right) \quad (18)$$

The class with the greatest probability is selected as the predicted class of an input vector in NB algorithm. The term $P(X = x)$ is identical for each class in NB classifier. Therefore, it can be ignored and the final classification formula is presented as (19) [53].

$$C^*(x) = \arg\max_i P(X = x|C = i) \times P(C = i) \quad (19)$$

Gaussian NB is a type of NB algorithm that works based on the assumption that the features of the dataset follow a Gaussian distribution. The probability density function of a conditional Gaussian distribution is presented as (20) [54], where μ_y and σ_y^2 are estimated by calculating the maximum likelihood.

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \quad (20)$$

4) RANDOM FOREST (RF)

Random Forest (RF) is an ensemble learning-based ML algorithm that creates an ensemble of Decision Trees (DTs) [55] with nodes and leaves through which a prediction is performed, given an input variable. In DT, each node contains a question about a single feature of an input item that is connected to a child node by a path, which represents an answer to the top/parent node. An item is classified throughout the path from the topmost node to a node without children, which is called a leaf. Each leaf represents the output class in DT. To construct a DT, an impurity measure called Gini index is utilized as demonstrated in (21) [56].

$$I = 1 - \sum_{i=1}^{m} p_i^2 \quad (21)$$

where p_i $(i = 1, 2, \ldots, m)$ is the fraction of the set of items related to an answer that belong to class i. To construct a DT, a question is selected that minimizes the weighted average impurity of the child nodes, associated with a parent node. In this regard, the possible k answers of a question that divide the set E into k subsets are considered to calculate the weighted average impurity, which is calculated as presented in (22) [56].

$$WI = \sum_{j=1}^{k} \left(\frac{|E_j|}{|E|}\right) I\left(E_j\right) \quad (22)$$

where E_j denotes the subset j of E. Moreover, $|E|$ and $|E_j|$ represent the size of E and E_j, respectively. In RF, each DT is established using a subset of the training data with a random number of selected features. Each DT predicts the class of an input as a base classifier and the output class is selected based on the majority voting of the predictions performed by DTs [55].

5) SUPPORT VECTOR MACHINE (SVM)

Support Vector Machine (SVM) is a classification algorithm that conducts classification by constructing a hyperplane that maximizes the margin between the classes. To train an SVM, a set of samples is required, where each sample is represented as a pair (x_i, y_i). x_i is representative of an input vector and y_i denotes its corresponding class label. Considering a training set $S = [(x_1, y_1), \ldots, (x_l, y_l)]$, where $x_i \in \mathbb{R}^d$ and $y_i \in [+1, -1]$. The separating hyperplanes for the two classes are calculated according to (23) [57].

$$y_i\left(w^T x_i + b\right) \geq 1 - \xi_i, \quad \forall i = 1, \ldots, l \quad (23)$$

where ξ_i are non-negative stack variables for data points that not completely satisfy the constraints. Using Lagrange dual, the final SVM optimization problem is presented as (24) to (26) [57].

$$\max_{\alpha_i} -\frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j K\left(x_i, x_j\right) + \sum_{i=1}^{l} \alpha_i \quad (24)$$

$$\sum_{i=1}^{l} \alpha_i y_i = 0 \quad (25)$$

$$0 \leq \alpha_i \leq C, \quad \forall i = 1, \ldots, l \quad (26)$$

where $K\left(x_i, x_j\right)$ denotes the kernel function, α_i represents the Lagrange multipliers, and $C \leq \infty$ is a penalty parameter. It is also notable that $\alpha_i \neq 0$ only for support vectors x_i. For a linearly separable case, a linear kernel is utilized as presented in (27) [57].

$$Linear : K\left(x_i, x_j\right) = x_i^T x_j \tag{27}$$

The optimal weight vector and Kuhn-Tucker condition are presented in (28) and (29), respectively [57].

$$w^* = \sum_{i=1}^{l} \alpha_i^* y_i x_i \tag{28}$$

$$\alpha_i^* \left[y_i \left(w^{*^T} x_i + b^*\right) - 1 + \xi_i\right] = 0 \tag{29}$$

The optimal hyperplane in SVM is calculated as demonstrated in (30) [57].

$$f\left(x\right) = \sum_{i=1}^{l} \alpha_i^* y_i K\left(x_i, x\right) + b^* \tag{30}$$

Finally, the decision function is defined as (31) [57].

$$C\left(x\right) = sign\left(\sum_{i=1}^{l} \alpha_i^* y_i K\left(x_i, x\right) + b^*\right) \tag{31}$$

Polynomial and RBF kernels are two of the most used kernel functions for non-linearly separable classes in SVM. The formulas of these two kernel functions are demonstrated in (32) and (33) [57], where r and γ represent the hyperparameters of polynomial and RBF kernels, respectively.

$$Polynomial : K\left(x_i, x_j\right) = \left(x_i^T x_j + 1\right)^r, \quad r \in \mathbb{Z}^+ \tag{32}$$

$$\alpha_i^* \left[y_i \left(w^{*^T} x_i + b^*\right) - 1 + \xi_i\right] = 0 \tag{33}$$

III. PROPOSED METHODOLOGY

The current research study proposed a new hybrid CNN model, using VGG16 and VGG19 networks as feature extractors to classify CXR images. In this hybrid CNN model, first, the features of CXR images are extracted using VGG16 and VGG19 networks. Then, the extracted features are concatenated and utilized to classify CXR images. In this regard, three classification approaches are employed. In the first classification approach, the proposed hybrid CNN model is utilized with FC layers and SoftMax activation function for the last FC layer to perform the classification. In the second classification approach, the concatenated features are deployed as the input of different ML classifiers. In the third classification approach, the classifiers obtained in the first and second approaches are assigned a weight and the probabilities are summed to calculate a new probability vector to classify CXR images accordingly. In this study, both VGG16 and VGG19 networks accept the same input images with a dimension of 224 × 224 pixels in parallel and combine the features of CXR images. Both VGG16 and VGG19 networks extract the features with a dimension of $7 \times 7 \times 512$ in their last layer of feature extraction, where the first two numbers (7) represent the width and height of the output feature map, respectively, and the third number (512) indicates the depth of the output feature map. The extracted features of VGG16 and VGG19 networks are concatenated and hence, an output with a dimension of $7 \times 7 \times 1024$ is achieved. These concatenated features are flattened to form a 1D vector, consisting of 50176 nodes.

To utilize FC layers in the proposed hybrid CNN model, the flattened features are connected to an FC layer containing 4096 nodes. This FC layer is connected to another FC layer with 4096 nodes and finally, the network is connected to two nodes, representing the normal and pneumonia classes. To get the most significant CXR image features, first, the image augmentation techniques are performed on the training set and the proposed hybrid CNN model with FC layers is trained for several epochs using SGD optimizer and the best weights that provide the highest test accuracy for classification of CXR images are saved. Conclusively, these weights can be deployed to get the most representative features of a new input image and predict its class with a significant accuracy. The architecture of the proposed hybrid model with FC layers and SoftMax activation function is depicted in Figure 5. The details of the proposed hybrid CNN model with FC layers, including the output shapes of the feature maps after each convolution/pooling layer, the number of parameters in each layer, and the total number of parameters are also presented in Table 1.

To employ ML classifiers in the proposed hybrid CNN model, first, the features of CXR images in the original training set are extracted using VGG16 and VGG19 networks, based on the saved optimized weights of the trained hybrid CNN model with FC layers. As these weights extract the most representative CXR image features, this weight utilization is conducted to construct robust ML classifiers with a significant classification performance. Then, these extracted features are concatenated and flattened to be used as the input of different ML classifiers, including KNN, LR, NB, RF, and SVM. The architecture of the proposed hybrid CNN model with ML classifiers is illustrated in Figure 6, where the three FC layers in the first classification approach have been replaced with ML classifiers. A pseudo-code of the proposed method is presented in Algorithm 1. According to Algorithm 1, the inputs are the training and test sets. Then, the hybrid CNN model with FC layers is trained using the training set and the performance is evaluated using the test set Accordingly, ML classifiers in the hybrid CNN model with ML classifiers are trained using the features extracted by the saved optimized weights and finally, the best-performing model is selected for making predictions.

A. DATA PRE-PROCESSING

The dataset used in this study was a public CXR dataset provided by Kermany *et al.* [58] which includes 5856 CXR images. These images include either normal or pneumonia classes. There are 1583 images fit in normal class and 4273 images fit to pneumonia class in this dataset. The pneumonia class involves images of viral and bacterial pneumonia, which are all categorized as pneumonia class. To use this

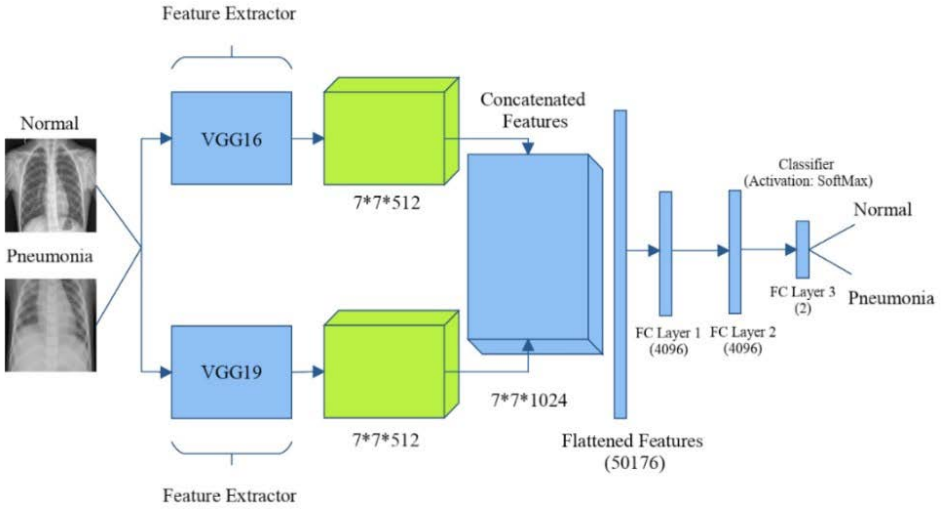


FIGURE 5. The architecture of the proposed hybrid CNN model with FC layers.

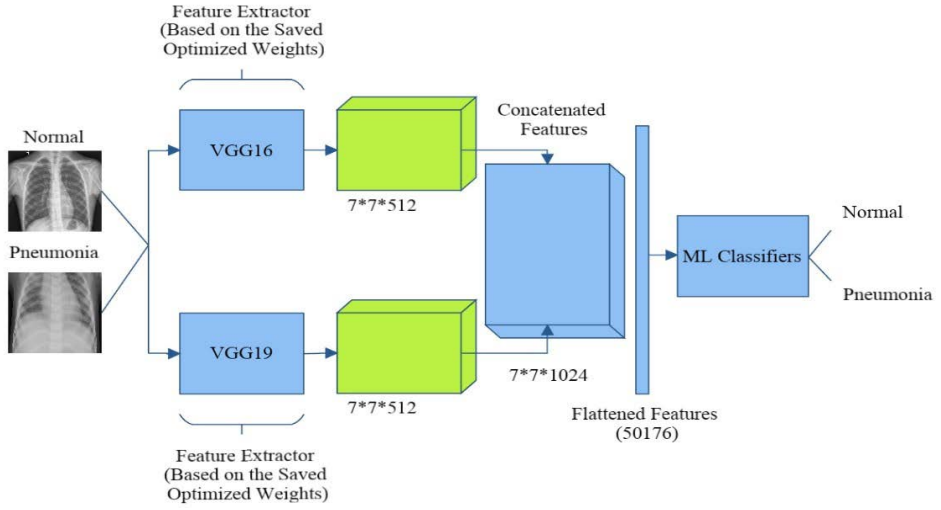


FIGURE 6. The architecture of the proposed hybrid CNN model with ML classifiers.

dataset for training the hybrid CNN model with FC layers/ML classifiers, 80% of the data was used for training, and the rest of 20% was utilized as the test data. Due to an unbalanced data, same proportion of images from each class was selected. In this regard, 80% of all images in each of normal and pneumonia classes were selected to create the training set. Similarly, 20% of images in each of normal and pneumonia classes were selected to create the test set. In total, the training set included 1267 images fit to normal class and 3419 images to pneumonia class. The test set consisted of 316 images fit to normal class and 854 images to pneumonia class. Examples of images fitting to normal and pneumonia images are shown in Figure 7. Furthermore, the statistics regarding the analyzed dataset are provided in Table 2.

To train the hybrid CNN model with FC layers/ML classifiers, some pre-processing operations were performed. VGG networks require input images of 224×224 pixels and hence, all the images in the training and test sets were resized to 224×224 pixels. All the images in the dataset contained pixel values in the range [0, 255] and hence, a 1./255 rescaling was employed to convert all the pixel values of CXR images to the range [0, 1]. These two pre-processing operations were the same for both classification approaches.

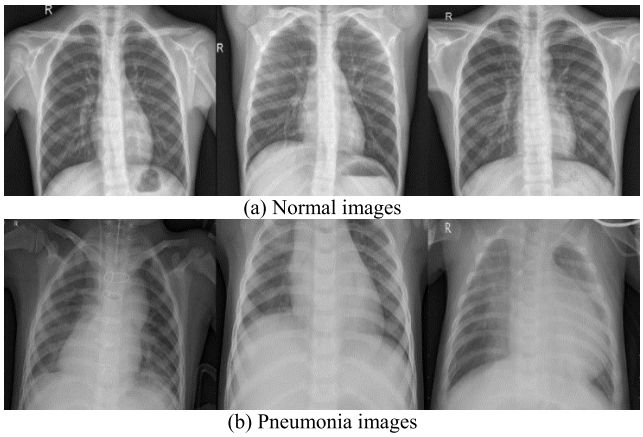
1) THE PROPOSED HYBRID CNN MODEL WITH FC LAYERS

To train the hybrid CNN model with FC layers, some additional preprocessing operations were conducted. Training a CNN requires a large dataset; otherwise, some parameters may not be appropriately estimated and accordingly, the CNN generalization will be poor. To tackle this problem, some image augmentation techniques were employed. The number of available images in the training set would increase by performing these image augmentation techniques on random samples of images. The used image augmentation techniques included flipping, zooming, and shearing. Examples of these augmentation techniques performed on an image in the dataset are portrayed in Figure 8.

TABLE 1. The details of the proposed hybrid CNN model with FC layers.

Layer Type*	Shape of the Feature Map (Width, Height, Depth)	Parameters
Input (Parallel for both VGG16 and VGG19 networks)	(224, 224, 3)	0
VGG16: Block 1-Convolution 1	(224, 224, 64)	1792
VGG16: Block 1-Convolution 2	(224, 224, 64)	36928
VGG16: Block 1-Max Pooling	(112, 112, 64)	0
VGG16: Block 2-Convolution 1	(112, 112, 128)	73856
VGG16: Block 2-Convolution 2	(112, 112, 128)	147584
VGG16: Block 2-Max Pooling	(56, 56, 128)	0
VGG16: Block 3-Convolution 1	(56, 56, 256)	295168
VGG16: Block 3-Convolution 2	(56, 56, 256)	590080
VGG16: Block 3-Convolution 3	(56, 56, 256)	590080
VGG16: Block 3-Max Pooling	(28, 28, 256)	0
VGG16: Block 4-Convolution 1	(28, 28, 512)	1180160
VGG16: Block 4-Convolution 2	(28, 28, 512)	2359808
VGG16: Block 4-Convolution 3	(28, 28, 512)	2359808
VGG16: Block 4-Max Pooling	(14, 14, 512)	0
VGG16: Block 5-Convolution 1	(14, 14, 512)	2359808
VGG16: Block 5-Convolution 2	(14, 14, 512)	2359808
VGG16: Block 5-Convolution 3	(14, 14, 512)	2359808
VGG16: Block 5-Max Pooling	(7, 7, 512)	0
VGG19: Block 1-Convolution 1	(224, 224, 64)	1792
VGG19: Block 1-Convolution 2	(224, 224, 64)	36928
VGG19: Block 1-Max Pooling	(112, 112, 64)	0
VGG19: Block 2-Convolution 1	(112, 112, 128)	73856
VGG19: Block 2-Convolution 2	(112, 112, 128)	147584
VGG19: Block 2-Max Pooling	(56, 56, 128)	0
VGG19: Block 3-Convolution 1	(56, 56, 256)	295168
VGG19: Block 3-Convolution 2	(56, 56, 256)	590080
VGG19: Block 3-Convolution 3	(56, 56, 256)	590080
VGG19: Block 3-Convolution 4	(56, 56, 256)	590080
VGG19: Block 3-Max Pooling	(28, 28, 256)	0
VGG19: Block 4-Convolution 1	(28, 28, 512)	1180160
VGG19: Block 4-Convolution 2	(28, 28, 512)	2359808
VGG19: Block 4-Convolution 3	(28, 28, 512)	2359808
VGG19: Block 4-Convolution 4	(28, 28, 512)	2359808
VGG19: Block 4-Max Pooling	(14, 14, 512)	0
VGG19: Block 5-Convolution 1	(14, 14, 512)	2359808
VGG19: Block 5-Convolution 2	(14, 14, 512)	2359808
VGG19: Block 5-Convolution 3	(14, 14, 512)	2359808
VGG19: Block 5-Convolution 4	(14, 14, 512)	2359808
VGG19: Block 5-Max Pooling	(7, 7, 512)	0
Concatenation	(7, 7, 1024)	0
Flattening	50176 (1D Vector)	0
FC Layer 1	4096 (1D Vector)	205524992
FC Layer 2	4096 (1D Vector)	16781312
FC Layer 3	2 (1D Vector)	8194
Total Parameters: 257,053,570		

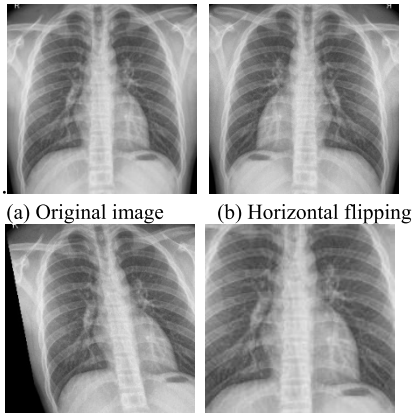
*The activation function of the convolution layers and the first two FC layers is ReLU. The activation function of FC layer 3 is SoftMax, which performs the predictions.



(a) Normal images

(b) Pneumonia images

FIGURE 7. Examples of images in the used dataset.



(a) Original image (b) Horizontal flipping

(c) Shearing (d) Zooming

FIGURE 8. The results of deploying various augmentation techniques on an image.

To prevent overfitting in training the proposed hybrid CNN model with FC layers, a dropout ratio of 0.5 was deployed after each FC layer during the training process. This means that 50% of the nodes are randomly dropped/deactivated at each update of the training phase in the network. These dropout layers are not available in the original VGG16 and VGG19 architectures, which can cause overfitting in some application cases and hence, the model performance and generalization would be degraded. To train the hybrid CNN model with FC layers, batches of 32 images were created and mini-batch GD optimizer with learning rate 0.01 was utilized to update the weights. The initial weights of the hybrid CNN model were set to ImageNet pre-trained weights, which were all updated during the training process. The loss function was set to binary CE and SoftMax activation function was deployed for the last FC layer to perform the classification. The hybrid CNN model was trained for 20 epochs, using the Keras library of Python on a Tesla K80 GPU with 12 GB RAM provided by Google Collaboratory Notebooks [59]. During the training process, the model checkpoint of the Keras library was utilized to save the weights of the best-performing model until a specific epoch based on the test accuracy. Therefore, the weights of the best-performing model were gained at the end of the training process. The hyperparameters, functions, and operations selected for training the hybrid CNN model with FC layers are presented in Table 3. In addition, the overall training process of the hybrid CNN model with FC layers is shown in Figure 9.

Algorithm 1 A Pseudo-Code of The Proposed Method

Input: Training and test sets.
Output: A model for detection of pneumonia cases.
(1) Train the hybrid CNN model with FC layers for several epochs using the training set.
(2) Save the weights that yield the highest test accuracy resulted in a specific epoch.
(3) Employ the saved optimized weights to extract the features of CXR images in the training set.
(4) Train ML classifiers, including KNN, LR, NB, RF, and SVM in the hybrid CNN model with ML classifiers.
(5) Use the hybrid CNN model with FC layers and LR, NB, and SVM classifiers to create an ensemble classifier and find the optimal weights.
(6) Calculate the test accuracy for each ML classifier and the ensemble classifier.
(7) Select the model with the highest test accuracy.
(8) Use the final best-performing model for making predictions.

TABLE 2. Description of the dataset.

Category	Training Set	Test Set
Normal	1267	316
Pneumonia	3419	854
Total	4686	1170
Percentage	80.02%	19.98%

2) THE PROPOSED HYBRID CNN MODEL WITH ML CLASSIFIERS

To train the proposed hybrid CNN model with ML classifiers, CXR image features have to be extracted in the first step. In this regard, the saved optimized weights of the trained hybrid CNN model with FC layers were loaded and utilized to train ML classifiers. This weight utilization was performed to extract the CXR image features that best represent the characteristics of the input CXR images to create robust ML classifiers. These features were flattened to be as the input of five ML classifiers, including KNN, LR, NB, RF, and SVM to classify CXR images. The overall training process of the hybrid CNN model with ML classifiers is portrayed in Figure 10.

To train the hybrid CNN model with KNN classifier, first, the hyperparameter K has to be defined. In this regard, different odd values for K in the range [3, 50] were considered and the KNN classifier was experimented using these K values to select the one that produces the highest test accuracy. Moreover, the distance measure was set to Euclidean distance. To classify a new input image, the KNN classifier uses an array of size 4686×50176, where 4686 stands for the number of extracted features of the images belonging to the

TABLE 3. The hyperparameters, functions, and operations selected for training the hybrid CNN model with FC layers.

Hyperparameters/Functions/Operations	Status
Learning Rate	0.01
Batch Size	32
Optimizer	SGD*
Loss Function	Binary CE
Epochs	20
Horizontal Flipping	Yes
Zoom Range	0.2
Shear Range	0.2
Rescaling	1./255

*To define mini-batch GD optimizer in the Keras library, the SGD optimizer is selected with a batch size bigger than 1 (32 in this paper).

training set and the KNN classifier calculates the distance of the extracted features of that input image from those 4686 instances to select the K nearest ones. To train the hybrid CNN model with RF classifier, an ensemble of 100 decision trees was employed, using Gini index as the impurity measure. To train the hybrid CNN model with SVM classifier, linear SVM, SVM with polynomial kernel, and SVM with RBF kernel (SVM-RBF) were utilized. For all these three SVM classifiers, a penalty parameter $C = 1$ was used. Moreover, the polynomial kernel was utilized with degree 3 ($r = 3$) to construct the SVM classifier with polynomial kernel. To construct the SVM-RBF classifier, the default value of the Keras library for γ was utilized, which is presented in (34).

$$\gamma = \frac{1}{N \times Var\,(X)} \tag{34}$$

where N is the number of training samples and Var (X) denotes the variance of all the training data points. For training the hybrid CNN model with NB classifier, Gaussian NB was used with 0.5 as the prior probability for both normal and pneumonia classes. Furthermore, a binary LR classifier with intercept was deployed to train the hybrid CNN model with LR classifier.

3) THE PROPOSED ENSMELBE CLASSIFIER

To train the proposed ensemble classifier, first, the trained hybrid CNN model with FC layers and ML classifiers has to be utilized to extract the features of CXR images in the test set. Then, a weight W_k is assigned to the probabilities obtained using the FC layers and ML classifiers. Accordingly, a weighted sum of class probabilities is calculated as presented in (35).

$$P_1 W_1 + P_2 W_2 + \ldots + P_k W_k = P_t \tag{35}$$

where P_k is a probability vector obtained by classifier k and W_k denotes its corresponding weight, which is in the range [0, 1]. Moreover, P_t represents the calculated probability vector. To create the proposed ensemble classifier, different values of W_k were tested to get the setting with the highest test accuracy. In total, there were six probability-based classifiers in this paper, including FC layers, linear SVM, SVM-RBF, polynomial SVM, LR, and NB, which were used to establish the proposed ensemble classifier. In this process, an array of size 1170*2 was obtained for each classifier, where 1170 indicates the number of images in the test set and 2 denotes the class probabilities for the normal and pneumonia classes. Accordingly, different set of weights were experimented for these six classifiers to obtain the best-performing ones in terms of the test accuracy.

B. FINDINGS AND RESULTS

In the current study, five performance metrics including accuracy, precision, recall, specificity, and F1-score have been reported to evaluate the performance of the models using the test set. The accuracy of a model demonstrates its overall effectiveness in identifying the actual positive and negative classes and precision indicates the effectiveness of the model in terms of predicting as belonging to positive class. Moreover, recall and specificity indicate the effectiveness of a model in identifying the actual positive and negative classes. In addition, F1-score provides an appropriate blend of precision and recall for evaluating the performance of a model. The values of these performance metrics were obtained using confusion matrices. The general form of the used confusion matrices in this study is presented in Figure 11. The formulas of the performance metrics are presented in (36) to (40).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{36}$$

$$Precision = \frac{TP}{TP + FP} \tag{37}$$

$$Recall = \frac{TP}{TP + FN} \tag{38}$$

$$Specificity = \frac{TN}{TN + FP} \tag{39}$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{40}$$

where TP, FP, TN, and FN are true positive, false positive, true negative, and false negative, respectively. TP shows the number of correctly classified images of pneumonia class, TN denotes the number of correctly classified images of normal class, FP stands for the number of wrongly classified images of normal class, and FN represents the number of images in pneumonia class, detected as belonging to normal class. The test loss was also used as another performance metric to assess the performance of the proposed hybrid CNN model with FC layers. Training accuracy and training loss were utilized to assess the performance of the model during the training process. As mentioned earlier, the selected loss function was binary CE.

The highest training accuracy in the proposed hybrid CNN model with FC layers is 97.78%, which was brought about in epoch 20. The trend of training accuracy with respect to epoch number is shown in Figure 12.

The highest test accuracy in the proposed hybrid CNN model with FC layers is 97.95%, which was appeared in

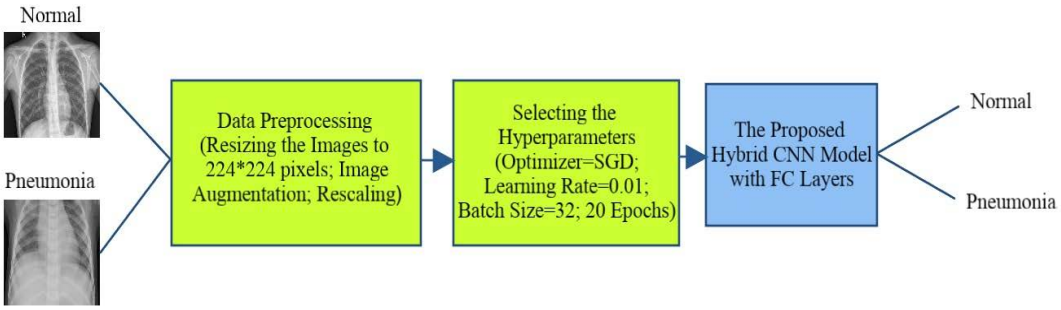


FIGURE 9. The training process of the proposed hybrid CNN model with FC layers.

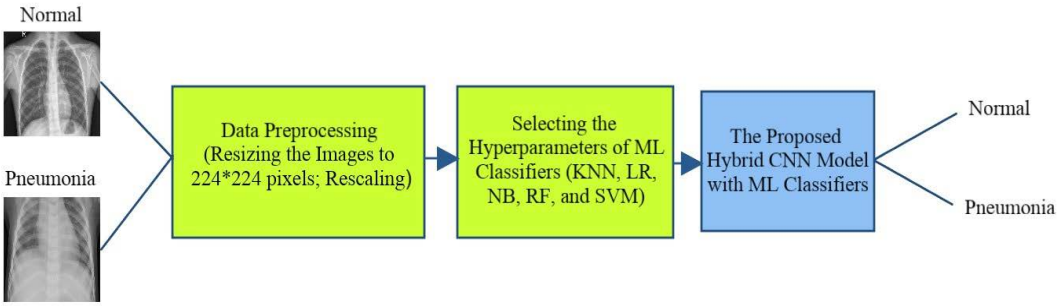


FIGURE 10. The training process of the proposed hybrid CNN model with FC layers.

epoch 16. The trend of test accuracy with respect to epoch number is plotted in Figure 13.

The trend of training loss with respect to epoch number for the binary classification of normal and pneumonia cases for the proposed hybrid CNN model with FC layers is shown in Figure 14. The minimum training loss is 0.0647, which was achieved in epochs 18 and 20.

The trend of test loss with respect to epoch number for the proposed hybrid CNN model with FC layers is shown in Figure 15. The minimum test loss is 0.0579, resulted in epoch 16, where the model achieved the highest test accuracy.

The training loss and test loss curves with respect to epoch number are illustrated in Figure 16. These curves indicate that the training loss and test loss values have been close to each other and there is not a huge gap between the training and test losses (0.0772 vs. 0.0579) in epoch 16 and in the overall training process of the model and hence, the model has not overfitted the training set during the training process.

The confusion matrix of the hybrid CNN model with FC layers is presented in Figure 17. The performance metrics values of the proposed hybrid CNN model with FC layers are also summarized in Table 4. This confusion matrix and the performance metrics values are based on epoch 16, where the model achieved the highest test accuracy. According to Table 4, the proposed hybrid CNN model with FC layers has a precision of 98.37%, a recall of 98.83%, a specificity of 95.57%, an F1-score of 98.60%, and an accuracy of 97.95%. Moreover, the test loss is equal to 0.0579. This test loss is the minimum test loss during the training process, which provided the highest test accuracy. To analyze the superiority of the proposed hybrid model with FC layers compared to

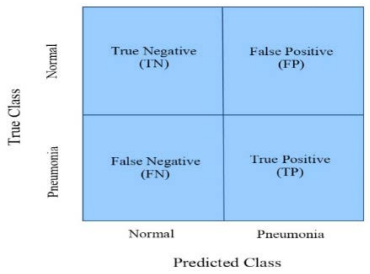


FIGURE 11. A general confusion matrix for evaluating the performance of the models.

individual VGG architectures, VGG16 and VGG19 architectures were trained separately. The confusion matrices of these two networks are presented in Figure 18 and Figure 19, respectively. Moreover, the performance metrics are provided in Table 5 and Table 6. The VGG16 and VGG19 have an accuracy of 97.78% and 97.69% respectively, which are both lower than that of the proposed hybrid CNN model with FC layers. The proposed model has also a better performance compared to both VGG16 and VGG19 networks in terms of precision, recall, specificity, F1-score, and test loss, which demonstrates the superiority of the proposed hybrid CNN model.

Furthermore, the confusion matrices of the proposed hybrid CNN model with ML classifiers are shown in Figure 20. The performance metrics of the hybrid CNN model with ML classifiers are also summarized in Table 7. According to the results provided in Table 7, the hybrid CNN model with KNN classifier has the highest precision, recall, specificity, F1-score, and accuracy compared to other models. These performance metrics values were achieved by a *K*

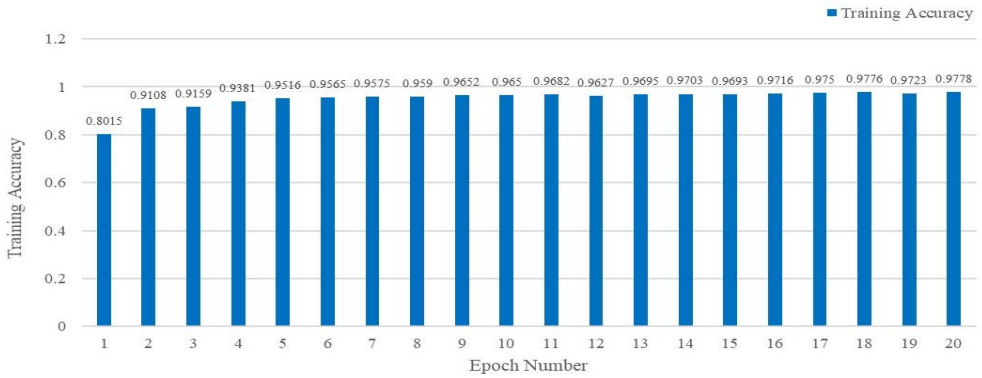


FIGURE 12. Training accuracy with respect to epoch number.

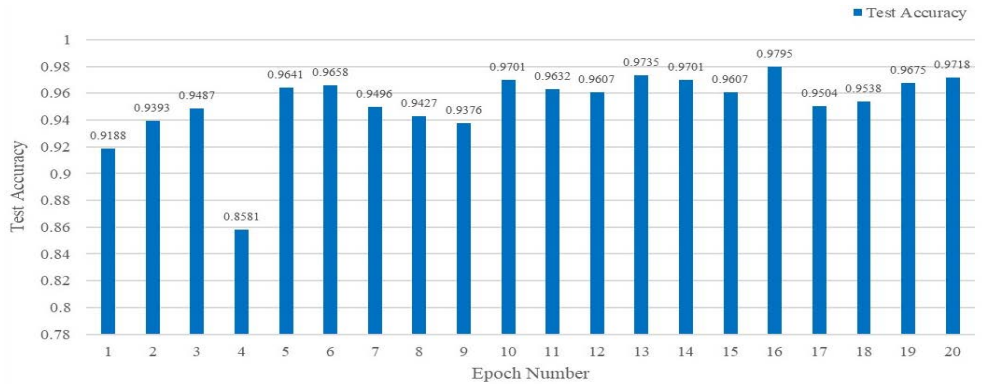


FIGURE 13. Test accuracy with respect to epoch number.

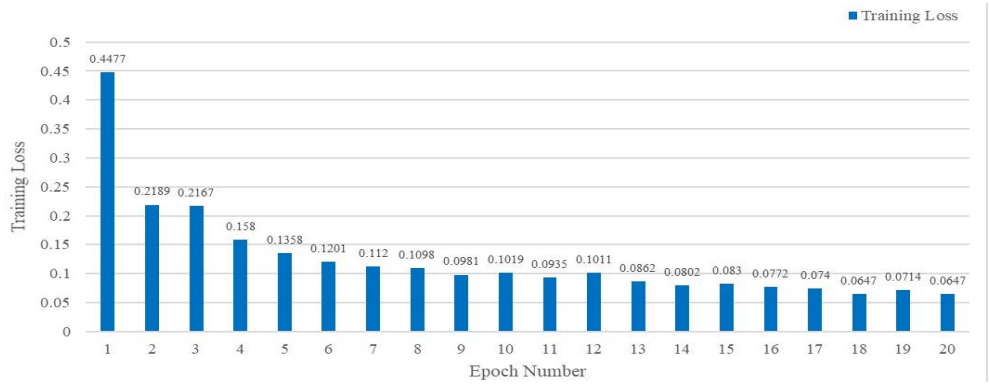


FIGURE 14. Training loss with respect to epoch number.

value of 13. The hybrid CNN model with KNN classifier has a precision of 98.83%, a recall of 99.18%, a specificity of 96.84%, an F1-score of 99.00%, and an accuracy of 98.55%, which is superior to the results of other ML classifiers. It is notable the proposed hybrid CNN model with SVM-RBF classifier has also a recall of 99.18%, which is equal to that of the proposed hybrid CNN model with KNN classifier; but in terms of other performance metrics, the KNN classifier has superior performance. Subsequently, the hybrid CNN model with SVM classifiers has an accuracy of 98.12%, 98.29%, and 98.29% for linear SVM, polynomial SVM, and SVM-RBF classifiers, respectively. The linear SVM and polynomial SVM classifiers have an equal precision of 98.60%; while it is 98.49% for SVM-RBF classifier. Similarly, the specificity for both linear SVM and polynomial SVM classifiers is equal to 96.20%, while it is 95.89% for SVM-RBF classifier. Moreover, the F1-score is equal to 98.71%, 98.83% and 98.83% for linear SVM, polynomial SVM, and SVM-RBF classifiers, respectively.

The hybrid CNN model with LR and RF classifiers has an accuracy of 98.03% and 97.69%, and precision of 98.48% and 97.80%, respectively. In terms of specificity, the RF classifier has a specificity of 93.99%, which is 1.9% lower than the specificity of LR classifier. Furthermore, the F1-scores are comparable for these two classifiers being equal to 98.65% and 98.43% for LR and RF, respectively. Although the

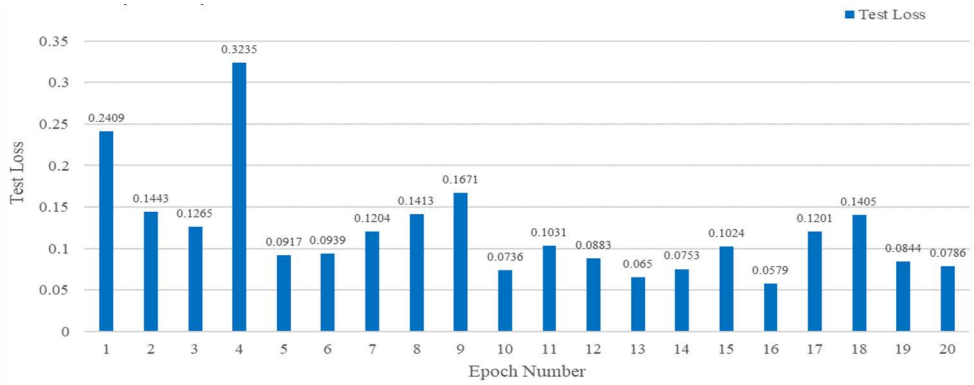


FIGURE 15. Test loss with respect to epoch number.

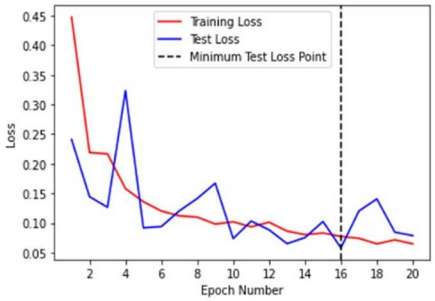


FIGURE 16. Training and test loss curves for the hybrid CNN model with FC layers.

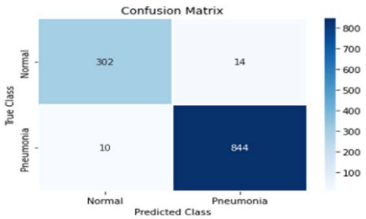


FIGURE 17. The confusion matrix of the hybrid CNN model with FC layers.

TABLE 4. The performance metrics of the hybrid CNN model with FC layers.

Performance Metric	Result (Value)
Precision	98.37%
Recall	98.83%
Specificity	95.57%
F1-score	98.60%
Accuracy	97.95%
Test Loss (Binary CE)	0.0579

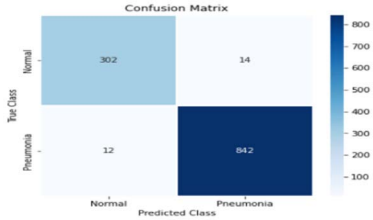


FIGURE 18. The confusion matrix of the VGG16 network.

precision, specificity, F1-score, and accuracy of LR and RF classifiers are lower that these performance metrics values for SVM classifiers, but in terms of recall, these two classifiers

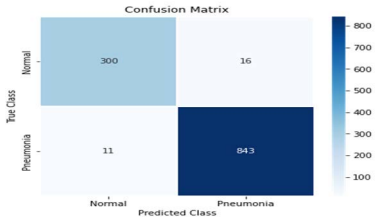


FIGURE 19. The confusion matrix of the VGG19 network.

TABLE 5. The performance metrics of the VGG16 network.

Performance Metric	Result (Value)
Precision	98.36%
Recall	98.59%
Specificity	95.57%
F1-score	98.47%
Accuracy	97.78%
Test Loss (Binary CE)	0.0599

TABLE 6. The performance metrics of the VGG19 network.

Performance Metric	Result (Value)
Precision	98.14%
Recall	98.71%
Specificity	94.94%
F1-score	98.42%
Accuracy	97.69%
Test Loss (Binary CE)	0.0589

performed on a par with SVM classifiers. The LR classifier has a recall of 98.83%, which is equal to the recall of linear SVM classifier. In addition, the RF classifier has a recall of 99.06%, which is equal to that of polynomial SVM classifier. The proposed hybrid CNN model with Gaussian NB classifier has a precision, recall, F1-score, and accuracy of more than 92%; but these performance metrics values are lower than those for other ML classifiers. Moreover, the Gaussian NB classifier has a specificity of 77.85%, which means that a high number of normal images are misclassified and detected as belonging to pneumonia class by this classifier.

The proposed ensemble classifier demonstrated the best results with a weight of 0.4 for SVM-RBF classifier and 0.1 for LR classifiers. Moreover, the weights of other classifiers, including FC layers, linear SVM, polynomial SVM, and NB classifiers were set to 0. The confusion matrix of the proposed ensemble classifier has been presented in Figure 21.

TABLE 7. The results of the hybrid CNN model with ML classifiers.

CNN Models with ML classifiers	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)	Accuracy (%)
CNN model with KNN classifier	**98.83***	**99.18**	**96.84**	**99.00**	**98.55**
CNN model with LR classifier	98.48	98.83	95.89	98.65	98.03
CNN model with Gaussian NB classifier	92.27	97.89	77.85	95.00	92.48
CNN model with RF classifier	97.80	99.06	93.99	98.43	97.69
CNN model with linear SVM Classifier	98.60	98.83	96.20	98.71	98.12
CNN model with polynomial SVM classifier	98.60	99.06	96.20	98.83	98.29
CNN model with SVM-RBF classifier	98.49	**99.18**	95.89	98.83	98.29

∗ **Bold** numbers indicate superior performance

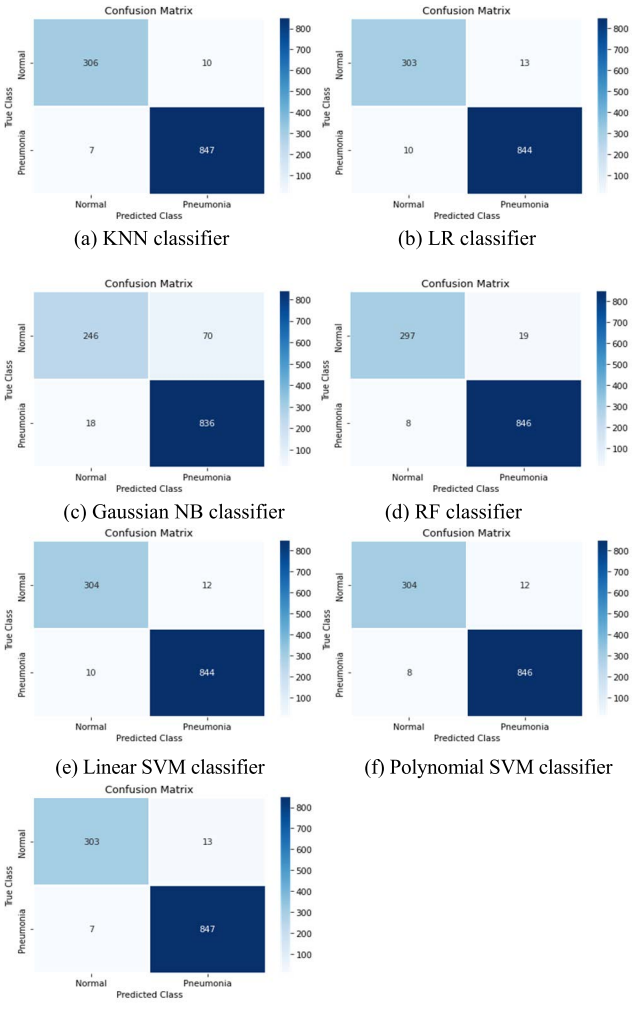


FIGURE 20. The confusion matrices of the hybrid CNN model with ML classifiers.

Moreover, the performance metrics have been analyzed in Table 8. According to Table 8, the proposed ensemble classifier has an accuracy of 98.55%, a precision of 98.72%, a recall of 99.30%, a specificity of 96.52%, and an F1-score of 99.01%. According to these results, the proposed ensemble classifier indicates a higher recall compared to the proposed hybrid CNN model with FC layers and all the ML classifiers. This ensemble classifier has also an accuracy equal to that of KNN classifier. Although the precision and specificity of the proposed hybrid CNN model with KKN classifier are

TABLE 8. The performance metrics of the proposed ensemble classifier.

Performance Metric	Result (Value)
Precision	98.72%
Recall	99.30%
Specificity	96.52%
F1-score	99.01%
Accuracy	98.55%

higher than those of the ensemble classifier, however, the recall of the ensemble classifier is higher than that of KKN classifier. Moreover, the F1-score of the ensemble classifier, which combines the precision and recall is slightly higher than the KNN classifier.

All the codes and calculations regarding the hybrid CNN model are provided in a public GitHub repository [60]. According to the final results, the proposed ensemble classifier outperforms the hybrid CNN model with FC layers and ML classifiers in terms of accuracy, recall, and F1-score. Although the hybrid CNN model with KNN classifier indicated an accuracy equal to that of the proposed ensemble classifier and has a higher precision and specificity, however, the ensemble classifier has a higher recall, which means that a greater number of pneumonia cases are detected by this model. Moreover, the F1-score of the ensemble classifier is slightly better than that of KNN classifier. Furthermore, the ensemble classifiers provide better generalization on the test set and reduce the classification bias and/or variance [61]. Therefore, the proposed ensemble classifier was selected as the final best model. The proposed ensemble classifier has been compared with existing studies in the literature regarding the detection of pneumonia in CXR images in Table 9. In these studies, a binary classification of normal and pneumonia images has been performed. According to Table 9, the proposed ensemble classifier in this study has a very high recall, compared to most of the recent studies. The accuracy, precision, and F1-score of the proposed ensemble classifier are higher than those of all the mentioned studies. It is also noteworthy that the test set used in this study was bigger than the test sets used in the compared studies, except in [18], and if more images for each class were used in the training process, these statistical measurements, as well as the generalization of the model would improve.

Consequently, the results of the study suggest that the proposed ensemble classifier can be utilized to aid radiologists in a time-efficient medical decision-making for pneumonia detection in CXR images with an accuracy of 98.55% and

TABLE 9. Comparison of the proposed model with existing studies.

Model	Total number of images	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)
Jain et al. [20]	5840	-	98	94	92.31
Stephen et al. [22]	5856	-	-	-	93.73
Kermany et al. [24]	5856	-	93.2	-	92.8
Moujahid et al. [64]	5856	91	97	94	96.81
Shah et al. [65]	5856	97.2	98.1	97.6	96.6
Manickam et al. [66]	5229	88.97	96.78	92.71	93.06
Fernandes et al. [17]	5232	96.4	97.9	97.2	96.4
Dey et al. [18]	7150	95.02	-	96.27	97.94
Rajaraman et al. [19]	5856	97.7	96.2	97.0	96.2
Sousa et al. [23]	5232	-	**99.7***	-	95.3
Liang and Zheng [21]	5856	89.1	96.7	92.7	90.5
Saraiva et al. [67]	5863	-	-	-	94.4
Chouhan et al. [68]	5232	93.28	99.62	-	96.39
Rahman et al. [25]	5247	97	99	98.1	98
Zhang et al. [26]	5786	94.41	90.82	92.58	96.07
Hashmi et al. [69]	5856	98.26	99.00	-	98.43
Mamalakis et al. [70]	5233	98.31	98.12	98.21	-
The proposed ensemble classifier	5856	**98.72**	99.30	**99.01**	**98.55**

* **Bold** numbers indicate superior performance.

a recall of 99.30%. In this regard, this model was deployed to implement a mobile phone web-delivered application of pneumonia detection. In order to implement this application, the saved optimized weights of the trained hybrid CNN model with FC layers (the first classification approach) were utilized to extract the features of an input CXR image and then the saved SVM-RBF and LR classifiers to create the proposed ensemble classifier and classify the input CXR image. The instructions regarding the implementation of this application are provided on a public Google collab repository [62]. The environment of the implemented application is demonstrated in Figure 22. This web application takes the input CXR image from the user. At that point, the features of CXR images are extracted using the saved optimized weights of the hybrid CNN model, which were gained in the first classification approach. Therefore, the most representative CXR image features are gained for the classification phase. Consequently, this web application employs the saved SVM-RBF and LR classifiers to create the proposed ensemble classifier and classify the input CXR image with an accuracy of 98.55%.

C. DISCUSSIONS, IMPLICATIONS, AND LIMITATIONS

Machine Learning (ML) and Deep Learning (DL) methods are one of the most effective tools that can assist physicians, clinicians, and radiologists in various medical applications and tasks. These methods can be utilized to establish CAD systems to detect diseases at an early stage in order to provide early treatments and reduce mortality rate in patients. Due to the rapid expansion of lung diseases in patients, there is an urgent need for early detection of lung diseases. As a noteworthy example, the delayed detection of widespread COVID-19 pneumonia can lead to a higher death probability and need for a highly-intensive care in patients, which can be prevented by early diagnosis [63]. In this regard, CAD systems can be effective for early detection of lung diseases. These systems can be much faster than an experienced radiologist in CXR image analysis. When there is a similarity

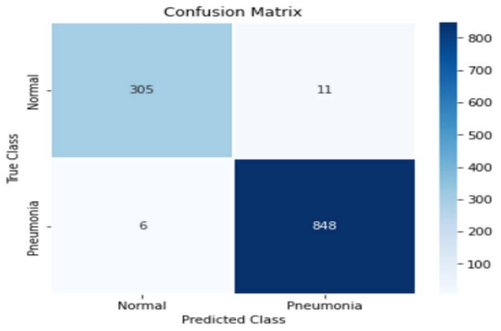


FIGURE 21. The confusion matrix of the proposed ensemble classifier.

between the features of diseases, CAD systems can effectively distinguish those features which can be a challenging task to radiologists. Therefore, CAD systems based on ML- and DL-based methods, and especially state-of-the-art CNN models can provide a more-informed and reliable framework for real-world medical decision-making by clinicians, radiologists, and healthcare experts.

There were some limitations in the current study. The first limitation of this study was the number of images available in the dataset. CNN models require large datasets with thousands of images to increase the generalization of the model. The generalization of the proposed hybrid CNN model with FC layers/ML classifiers in this study would improve by using a larger dataset with more images for both the normal class and pneumonia class in the training process. Moreover, the performance of the proposed hybrid CNN model with FC layers/ML classifiers would also improve by deploying larger datasets. The second limitation is that there is a need for developing techniques for specifying the regions of infection in CXR images, combined with the proposed models to assist radiologists for a more informed medical decision-making.

For future studies, using VGG variants for features extraction which is a long-established way to build good baseline solutions is encouraged. In fact, the whole image level binary classification may not attract a good amount of audience.

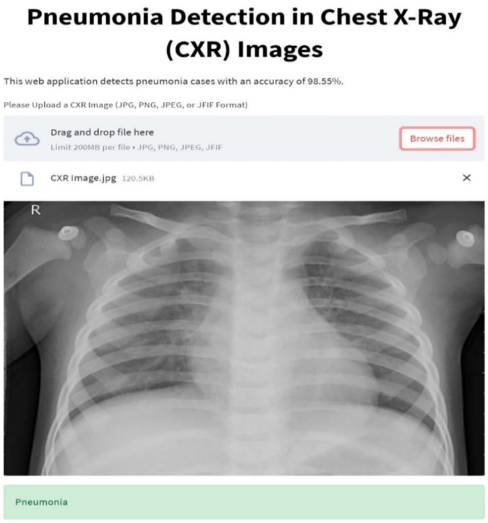


FIGURE 22. The environment of the implemented mobile phone web application for pneumonia detection.

In addition, moving toward the multi-class tasks and expanding the dataset, are encouraged for future research.

IV. CONCLUSION

Pneumonia is a significant cause of death in children and adults in the world. Pneumonia can be prevented with modest treatments and cured with low-cost, low-tech medication and care. In this study, a new hybrid VGG-based CNN model was proposed to detect pneumonia in CXR images. In this regard, three classification approaches were utilized. In the first classification approach, the hybrid CNN model was deployed with FC layers and was trained to gain the weights that produce the highest classification accuracy. In the second classification approach, the saved optimized weights were employed to extract the most representative CXR image features and five ML classifiers, including KNN, LR, NB, RF, and SVM were utilized to classify CXR images. In the third classification approach, an ensemble classifier using the trained classifiers in the first and second classification approaches was created. The best-performing model was the proposed ensemble classifier using a weight of 0.4 for SVM-RBF classifier and a weight of 0.1 for LR classifier, which achieved an accuracy of 98.55%, a precision of 98.72%, a recall of 99.30%, and an F1-score of 99.01%. This model had the best performance in terms of all performance metrics, compared to the hybrid model with FC layers and all the ML classifiers. Moreover, this model had a superior performance, compared to existing algorithm within the literature regarding the detection of pneumonia in CXR images. Ultimately, a mobile phone web-delivered application of pneumonia detection was designed using the proposed ensemble classifier, which can assist radiologists in pneumonia detection with a significant accuracy.

Even though this research study has accomplished the research intentions, some limitations still exist. Hence, recommendations for future directions, developments, and applications of the current study can be summarized and demonstrated as follows. First, the performance of the proposed hybrid CNN model can be further improved by developing new robust classifiers for the classification infection areas in CXR images using the radiologists' knowledge and DL-based methods, along with the proposed hybrid CNN model, a new framework can be established to assist radiologists for a more informed and reliable medical decision-making. Third, the performance of the hybrid CNN model with FC layers can be improved by fine-tuning the hyperparameters of the model, including the learning rate and optimizer. Fourth, the hybrid CNN model with FC layers can be utilized to create a weighted ensemble CNN model using the pre-trained CNN models to increase the classification accuracy. Fifth, the hybrid CNN model can be deployed for classification of COVID-19 pneumonia and other viral pneumonia types. Sixth, the hybrid CNN model can be applied to other medical decision-making problems, such as classification of other lung diseases and infections.

ACKNOWLEDGMENT

Conflict of Interests: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

REFERENCES

[1] World Health Organization. *Pneumonia*. Accessed: Feb. 20, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/pneumonia

[2] A. Torres, C. Cilloniz, M. S. Niederman, R. Menéndez, J. D. Chalmers, R. G. Wunderink, and T. van der Poll, "Pneumonia," *Nature Rev. Disease Primers*, vol. 7, no. 1, p. 25, Dec. 2021, doi: 10.1038/s41572-021-00259-0.

[3] Save the Children. *Pneumonia to Kill Nearly 11 Million Children by 2030*. Accessed: Feb. 20, 2022. [Online]. Available: https://www.savethechildren.org/us/about-us/media-and-news/2018-press-releases/pneumonia-to-kill-nearly-11-million-children-by-2030

[4] F. W. Arnold, A. M. R. Vega, V. Salunkhe, S. Furmanek, C. Furman, L. Morton, A. Faul, P. Yankeelov, and J. A. Ramirez, "Older adults hospitalized for pneumonia in the United States: Incidence, epidemiology, and outcomes," *J. Amer. Geriatrics Soc.*, vol. 68, no. 5, pp. 1007–1014, May 2020, doi: 10.1111/jgs.16327.

[5] World Health Organization. *Household Air Pollution and Health*. Accessed: Feb. 20, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/household-air-pollution-and-health

[6] A. H. Attaway, R. G. Scheraga, A. Bhimraj, M. Biehl, and U. Hatipoğlu, "Severe COVID-19 pneumonia: Pathogenesis and clinical management," *BMJ*, vol. 372, p. n436, Mar. 2021, doi: 10.1136/bmj.n436.

[7] O. Ruuskanen, E. Lahti, L. C. Jennings, and D. R. Murdoch, "Viral pneumonia," *Lancet*, vol. 377, no. 9773, pp. 1264–1275, Apr. 2011, doi: 10.1016/s0140-6736(10)61459-6.

[8] RadiologyInfo. *Pneumonia*. Accessed: Feb. 20, 2022. [Online]. Available: https://www.radiologyinfo.org/en/info/pneumonia

[9] W. H. Self, D. M. Courtney, C. D. McNaughton, R. G. Wunderink, and J. A. Kline, "High discordance of chest X-ray and computed tomography for detection of pulmonary opacities in ED patients: Implications for diagnosing pneumonia," *Amer. J. Emergency Med.*, vol. 31, no. 2, pp. 401–405, Feb. 2013, doi: 10.1016/j.ajem.2012.08.041.

[10] K. Cherney. MRI vs. X-ray: What you need to know. Healthline. Accessed: Feb. 20, 2022. [Online]. Available: https://www.healthline.com/health/mri-vs-xray

[11] T. P. Htun, Y. Sun, H. L. Chua, and J. Pang, "Clinical features for diagnosis of pneumonia among adults in primary care setting: A systematic and meta-review," *Sci. Rep.*, vol. 9, no. 1, p. 7600, Dec. 2019, doi: 10.1038/s41598-019-44145-y.

[12] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: Review, opportunities and challenges," *Briefings Bioinf.*, vol. 19, no. 6, pp. 1236–1246, Nov. 2018, doi: 10.1093/bib/bbx044.

[13] H. Chan, L. M. Hadjiiski, and R. K. Samala, "Computer-aided diagnosis in the era of deep learning," *Med. Phys.*, vol. 47, no. 5, pp. e218–e227, May 2020, doi: 10.1002/mp.13764.

[14] P. Rajpurkar *et al.*, "Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists," *PLOS Med.*, vol. 15, no. 11, Nov. 2018, Art. no. e1002686, doi: 10.1371/journal.pmed.1002686.

[15] J. Seah *et al.*, "Effect of a comprehensive deep-learning model on the accuracy of chest X-ray interpretation by radiologists: A retrospective, multireader multicase study," *The Lancet Digital Health*, vol. 3, pp. e496–e506, Aug. 2021, doi: 10.1016/S2589-7500(21)00106-0.

[16] A. Hosny, C. Parmar, J. Quackenbush, L. H. Schwartz, and H. J. W. L. Aerts, "Artificial intelligence in radiology," *Nature Rev. Cancer*, vol. 18, no. 8, pp. 500–510, 2018, doi: 10.1038/s41568-018-0016-5.

[17] V. Fernandes, G. B. Junior, A. C. de Paiva, A. C. Silva, and M. Gattass, "Bayesian convolutional neural network estimation for pediatric pneumonia detection and diagnosis," *Comput. Methods Programs Biomed.*, vol. 208, Sep. 2021, Art. no. 106259, doi: 10.1016/j.cmpb.2021.106259.

[18] N. Dey, Y.-D. Zhang, V. Rajinikanth, R. Pugalenthi, and N. S. M. Raja, "Customized VGG19 architecture for pneumonia detection in chest X-rays," *Pattern Recognit. Lett.*, vol. 143, pp. 67–74, Mar. 2021, doi: 10.1016/j.patrec.2020.12.010.

[19] S. Rajaraman, S. Candemir, I. Kim, G. Thoma, and S. Antani, "Visualization and interpretation of convolutional neural network predictions in detecting pneumonia in pediatric chest radiographs," *Appl. Sci.*, vol. 8, no. 10, p. 1715, Sep. 2018, doi: 10.3390/app8101715.

[20] R. Jain, P. Nagrath, G. Kataria, V. S. Kaushik, and D. J. Hemanth, "Pneumonia detection in chest X-ray images using convolutional neural networks and transfer learning," *Measurement*, vol. 165, Dec. 2020, Art. no. 108046, doi: 10.1016/j.measurement.2020.108046.

[21] G. Liang and L. Zheng, "A transfer learning method with deep residual network for pediatric pneumonia diagnosis," *Comput. Methods Programs Biomed.*, vol. 187, Apr. 2020, Art. no. 104964, doi: 10.1016/j.cmpb.2019.06.023.

[22] O. Stephen, M. Sain, U. J. Maduh, and D.-U. Jeong, "An efficient deep learning approach to pneumonia classification in healthcare," *J. Healthcare Eng.*, vol. 2019, Mar. 2019, Art. no. 4180949, doi: 10.1155/2019/4180949.

[23] G. G. B. Sousa, V. R. M. Fernandes, and A. C. de Paiva, "Optimized deep learning architecture for the diagnosis of pneumonia through chest X-rays," in *Image Analysis and Recognition*, vol. 11663, F. Karray, A. Campilho, and A. Yu, Eds. Cham, Switzerland: Springer, 2019, pp. 353–361.

[24] D. S. Kermany *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018, doi: 10.1016/j.cell.2018.02.010.

[25] T. Rahman, M. E. H. Chowdhury, A. Khandakar, K. R. Islam, K. F. Islam, Z. B. Mahbub, M. A. Kadir, and S. Kashem, "Transfer learning with deep convolutional neural network (CNN) for pneumonia detection using chest X-ray," *Appl. Sci.*, vol. 10, no. 9, p. 3233, May 2020, doi: 10.3390/app10093233.

[26] D. Zhang, F. Ren, Y. Li, L. Na, and Y. Ma, "Pneumonia detection from chest X-ray images based on convolutional neural network," *Electronics*, vol. 10, no. 13, p. 1512, Jun. 2021, doi: 10.3390/electronics10131512.

[27] V. Gavrishchaka, Z. Yang, R. Miao, and O. Senyukova, "Advantages of hybrid deep learning frameworks in applications with limited data," *Int. J. Mach. Learn. Comput.*, vol. 8, pp. 549–558, Dec. 2018, doi: 10.18178/ijmlc.2018.8.6.744.

[28] S. Ardabili, A. Mosavi, and A. Varkonyi-Koczy, "Advances in machine learning modeling reviewing hybrid and ensemble methods," in *Engineering for Sustainable Future*. Springer, 2020, pp. 215–227.

[29] W. Saad, W. A. Shalaby, M. Shokair, F. A. El-Samie, M. Dessouky, and E. Abdellatef, "COVID-19 classification using deep feature concatenation technique," *J. Ambient Intell. Hum. Comput.*, vol. 13, no. 4, pp. 2025–2043, Apr. 2022, doi: 10.1007/s12652-021-02967-7.

[30] N. Noreen, S. Palaniappan, A. Qayyum, I. Ahmad, M. Imran, and M. Shoaib, "A deep learning model based on concatenation approach for the diagnosis of brain tumor," *IEEE Access*, vol. 8, pp. 55135–55144, 2020, doi: 10.1109/ACCESS.2020.2978629.

[31] J. Penm, B. Chaar, R. Moles, and J. Penm, "Predicting ASX health care stock index movements after the recent financial crisis using patterned neural networks," in *Rethinking Valuation and Pricing Models*, C. S. Wehn, C. Hoppe, and G. N. Gregoriou Eds. New York, NY, USA: Academic, 2013, ch. 37, pp. 599–610.

[32] A. Urso, A. Fiannaca, M. La Rosa, V. Ravì, and R. Rizzo, "Data mining: Classification and prediction," in *Encyclopedia of Bioinformatics and Computational Biology*, S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, Eds. New York, NY, USA: Academic, 2019, pp. 384–402.

[33] S. Nisha and N. Meeral, "Applications of deep learning in biomedical engineering," in *Handbook of Deep Learning in Biomedical Engineering*. Springer, 2020, pp. 245–270.

[34] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: An overview and application in radiology," *Insights Imag.*, vol. 9, pp. 611–629, Aug. 2018, doi: 10.1007/s13244-018-0639-9.

[35] J. Ker, L. Wang, J. Rao, and T. Lim, "Deep learning applications in medical image analysis," *IEEE Access*, vol. 6, pp. 9375–9389, 2018, doi: 10.1109/ACCESS.2017.2788044.

[36] C. Zhou, J. Song, S. Zhou, Z. Zhang, and J. Xing, "COVID-19 detection based on image regrouping and resnet-SVM using chest X-ray images," *IEEE Access*, vol. 9, pp. 81902–81912, 2021, doi: 10.1109/ACCESS.2021.3086229.

[37] K. Balaji and K. Lavanya, "Medical image analysis with deep neural networks," in *Deep Learning and Parallel Computing Environment for Bioengineering Systems*, A. K. Sangaiah, Ed. New York, NY, USA: Academic, 2019, ch. 5, pp. 75–97.

[38] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," 2016, *arXiv:1603.07285*.

[39] E. Fathi and B. M. Shoja, "Deep neural networks for natural language processing," in *Handbook of Statistics*, vol. 38, V. N. Gudivada and C. R. Rao, Eds. Amsterdam, The Netherlands: Elsevier, 2018, ch. 9, pp. 229–316.

[40] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.

[41] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, "The marginal value of adaptive gradient methods in machine learning," in *Proc. NIPS*, 2017, pp. 1–11.

[42] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[43] P. Mehta, M. Bukov, C.-H. Wang, A. G. R. Day, C. Richardson, C. K. Fisher, and D. J. Schwab, "A high-bias, low-variance introduction to machine learning for physicists," *Phys. Rep.*, vol. 810, pp. 1–124, May 2019, doi: 10.1016/j.physrep.2019.03.001.

[44] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255, doi: 10.1109/CVPR.2009.5206848.

[45] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

[46] H. Zhang, L. Zhang, and Y. Jiang, "Overfitting and underfitting analysis for deep learning based end-to-end communication systems," in *Proc. 11th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2019, pp. 1–6, doi: 10.1109/WCSP.2019.8927876.

[47] C. M. Bishop, *Pattern Recognition and Machine Learning* (Information Science and Statistics). New York, NY, USA: Springer, 2006, p. 738.

[48] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 56, pp. 1929–1958, 2014.

[49] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, Dec. 2019, Art. no. 60, doi: 10.1186/s40537-019-0197-0.

[50] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, 2007, doi: 10.1007/s10115-007-0114-2.

[51] J. Han, M. Kamber, and J. Pei, "Getting to know your data," in *Data Mining*, 3rd ed., J. Han, M. Kamber, and J. Pei, Eds. Boston, MA, USA: Morgan Kaufmann, 2012, ch. 2, pp. 39–82.

[52] J. I. E. Hoffman, "Logistic Regression," in *Biostatistics for Medical and Biomedical Practitioners*, J. I. E. Hoffman, Ed. New York, NY, USA: Academic, 2015, ch. 33, pp. 601–611.

[53] I. Rish, "An empirical study of the Naïve Bayes classifier," in *Proc. IJCAI Work Empirical Methods Artif. Intell.*, vol. 3, Jan. 2001, pp. 41–46.

[54] W. Lou, X. Wang, F. Chen, Y. Chen, B. Jiang, and H. Zhang, "Sequence based prediction of DNA-binding proteins based on hybrid feature selection using random forest and Gaussian Naïve Bayes," *PLoS ONE*, vol. 9, no. 1, Jan. 2014, Art. no. e86703, doi: 10.1371/journal.pone.0086703.
[55] K. Fawagreh, M. M. Gaber, and E. Elyan, "Random forests: From early developments to recent advancements," *Syst. Sci. Control Eng.*, vol. 2, no. 1, pp. 602–609, Dec. 2014, doi: 10.1080/21642583.2014.956265.
[56] C. Kingsford and S. L. Salzberg, "What are decision trees?" *Nature Biotechnol.*, vol. 26, no. 9, pp. 1011–1013, Sep. 2008, doi: 10.1038/nbt0908-1011.
[57] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, Sep. 2020, doi: 10.1016/j.neucom.2019.10.118.
[58] D. S. Kermany, K. Zhang, and M. Goldbaum. *Labeled Optical Coherence Tomography (OCT) and Chest X-ray Images for Classification*. Accessed: Feb. 20, 2022. [Online]. Available: https://data.mendeley.com/datasets/rscbjbr9sj/2
[59] Google. *Google Collaboratory Notebooks*. Accessed: Feb. 20, 2022. [Online]. Available: https://colab.research.google.com/notebooks/
[60] M. Yaseliani, A. Z. Hamadani, A. I. Maghsoodi, and A. Mosavi. *Pneumonia Detection in Chest X-ray Images Using a Hybrid Deep Convolutional Neural Network Based on Two Parallel Visual Geometry Group Architectures and Machine Learning Classifiers*. Accessed: Feb. 20, 2022. [Online]. Available: https://github.com/mohammadyaseliani/Pneumonia-Paper
[61] M. Re and G. Valentini, *Ensemble Methods: A Review*. London U.K.: CRC Press, 2012, pp. 563–594.
[62] M. Yaseliani, A. Z. Hamadani, A. I. Maghsoodi, and A. Mosavi. *A Web Application for Pneumonia Detection in Chest X-Ray Images Using a Hybrid Deep Convolutional Neural Network Based on Two Parallel Visual Geometry Group Architectures and K-Nearest Neighbors Classifier*. Accessed: Feb. 20, 2022. [Online]. Available: https://colab.research.google.com/github/mohammadyaseliani/Pneumonia-Paper/blob/main/Web%20Application.ipynb
[63] D. Goyal, M. Inada-Kim, F. Mansab, A. Iqbal, B. McKinstry, A. P. Naasan, C. Millar, S. Thomas, S. Bhatti, D. Lasserson, and D. Burke, "Improving the early identification of COVID-19 pneumonia: A narrative review," *BMJ Open Respiratory Res.*, vol. 8, no. 1, Nov. 2021, Art. no. e000911, doi: 10.1136/bmjresp-2021-000911.
[64] H. Moujahid, B. Cherradi, O. E. Gannour, L. Bahatti, O. Terrada, and S. Hamida, "Convolutional neural network based classification of patients with pneumonia using X-ray lung images," *Adv. Sci., Technol. Eng. Syst. J.*, vol. 5, no. 5, pp. 167–175, 2020, doi: 10.25046/aj050522.
[65] U. Shah, A. Abd-Alrazeq, T. Alam, M. Househ, and Z. Shah, "An efficient method to predict pneumonia from chest X-rays using deep learning approach," *Stud. Health Technol. Inf.*, vol. 272, pp. 457–460, Jun. 2020, doi: 10.3233/shti200594.
[66] A. Manickam, J. Jiang, Y. Zhou, A. Sagar, R. Soundrapandiyan, and R. D. J. Samuel, "Automated pneumonia detection on chest X-ray images: A deep learning approach with different optimizers and transfer learning architectures," *Measurement*, vol. 184, Nov. 2021, Art. no. 109953, doi: 10.1016/j.measurement.2021.109953.
[67] A. Saraiva, D. Santos, N. Costa, J. Sousa, N. Ferreira, A. Valente, and S. Soares, "Models of learning to classify X-ray images for the detection of pneumonia using neural networks," in *Proc. BIOIMAGING*, 2019, pp. 76–83, doi: 10.5220/0007346600760083.
[68] V. Chouhan, S. K. Singh, A. Khamparia, D. Gupta, P. Tiwari, C. Moreira, R. Damaševičius, and V. H. C. de Albuquerque, "A novel transfer learning based approach for pneumonia detection in chest X-ray images," *Appl. Sci.*, vol. 10, no. 2, p. 559, Jan. 2020, doi: 10.3390/app10020559.
[69] M. F. Hashmi, S. Katiyar, A. G. Keskar, N. D. Bokde, and Z. W. Geem, "Efficient pneumonia detection in chest Xray images using deep transfer learning," *Diagnostics*, vol. 10, no. 6, p. 417, Jun. 2020, doi: 10.3390/diagnostics10060417.
[70] M. Mamalakis, A. J. Swift, B. Vorselaars, S. Ray, S. Weeks, W. Ding, R. H. Clayton, L. S. Mackenzie, and A. Banerjee, "DenResCov-19: A deep transfer learning network for robust automatic classification of COVID-19, pneumonia, and tuberculosis from X-rays," *Comput. Med. Imag. Graph.*, vol. 94, Dec. 2021, Art. no. 102008, doi: 10.1016/j.compmedimag.2021.102008.

MOHAMMAD YASELIANI received the B.Sc. degree in industrial engineering (IE) from the Sharif University of Technology (SUT). He is currently pursuing the M.Sc. degree in IE with the Isfahan University of Technology (IUT). His research interests include deep learning (DL), machine learning (ML), and medical image processing (MIP).

ALI ZEINAL HAMADANI received the M.Sc. degree in applied statistics from the University of Sussex, Brighton, U.K., and the Ph.D. degree in industrial technology from the University of Bradford, Bradford, U.K. He is currently working as a Professor with the Department of Industrial Engineering, Isfahan University of Technology, Isfahan, Iran. He has authored many research articles published in highly ranked journals.

ABTIN IJADI MAGHSOODI received the B.Sc. degree in industrial engineering and the M.Sc. degree in industrial and systems engineering from Islamic Azad University, Iran, in 2015 and 2018, respectively. He is currently pursuing the Ph.D. degree in operations and supply chain management with the Department of Information Systems and Operations Management (ISOM), University of Auckland. He has authored several research articles published in highly ranked journals, including *Computers and Industrial Engineering*, *European Research on Management and Business Economics*, *Knowledge-Based Systems*, *Archives of Civil and Mechanical Engineering*, and *Applied Soft Computing*.

AMIR MOSAVI received the graduate degree from Kingston University London, U.K., and the Ph.D. degree in applied informatics. He was a Senior Research Fellow with Oxford Brookes University and the Queensland University of Technology. He is an Associate Professor of artificial intelligence and machine learning. He is a Data Scientist for climate change, sustainability, and hazard prediction. He was a recipient of the Alexander von Humboldt Award, the Green-Talent Award, the UNESCO Young Scientist Award, the Alain Bensoussan Fellowship, the GO STYRIA Award, the Estonian Dora Plus Grant, the Estophilus Scholarship, the Future Talent TU Darmstadt, the World Academy of Sciences UNESCO Award, the Marie Curie Award, the Endeavour-Australia Leadership Award, the Talented Young Scientist Award, the Slovak National Research Award, and the European Research Consortium for Informatics and Mathematics Fellowship.