Available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/bbe

Original Research Article

Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: A machine learning based approach

Sara Hosseinzadeh Kassani[a,*], Peyman Hosseinzadeh Kassani[b], Michal J. Wesolowski[c], Kevin A. Schneider[a], Ralph Deters[a]

[a] Department of Computer Science, University of Saskatchewan, Canada
[b] Department of Neurology and Neurological, University of Stanford, USA
[c] Department of Medical Imaging, University of Saskatchewan, Canada

ARTICLE INFO

Article history:
Received 22 April 2020
Received in revised form
17 January 2021
Accepted 13 May 2021
Available online 5 June 2021

Keywords:
Coronavirus disease
Lung opacity
Computer-aided diagnosis
Deep learning
Feature extraction
Transfer learning

ABSTRACT

The newly identified Coronavirus pneumonia, subsequently termed COVID-19, is highly transmittable and pathogenic with no clinically approved antiviral drug or vaccine available for treatment. The most common symptoms of COVID-19 are dry cough, sore throat, and fever. Symptoms can progress to a severe form of pneumonia with critical complications, including septic shock, pulmonary edema, acute respiratory distress syndrome and multi-organ failure. While medical imaging is not currently recommended in Canada for primary diagnosis of COVID-19, computer-aided diagnosis systems could assist in the early detection of COVID-19 abnormalities and help to monitor the progression of the disease, potentially reduce mortality rates. In this study, we compare popular deep learning-based feature extraction frameworks for automatic COVID-19 classification. To obtain the most accurate feature, which is an essential component of learning, MobileNet, DenseNet, Xception, ResNet, InceptionV3, InceptionResNetV2, VGGNet, NASNet were chosen amongst a pool of deep convolutional neural networks. The extracted features were then fed into several machine learning classifiers to classify subjects as either a case of COVID-19 or a control. This approach avoided task-specific data pre-processing methods to support a better generalization ability for unseen data. The performance of the proposed method was validated on a publicly available COVID-19 dataset of chest X-ray and CT images. The DenseNet121 feature extractor with Bagging tree classifier achieved the best performance with 99% classification accuracy. The second-best learner was a hybrid of the a ResNet50 feature extractor trained by LightGBM with an accuracy of 98%.

[*] Corresponding author at: Department of Computer Science, University of Saskatchewan, Saskatoon, Canada.
E-mail address: sara.kassani@usask.ca (S.H. Kassani).

https://doi.org/10.1016/j.bbe.2021.05.013

1. Introduction

A series of pneumonia cases of unknown etiology occurred in December 2019, in Wuhan, Hubei province, China. On December 31, 2019, 27 unexplained cases of pneumonia were identified and found to be associated with so called "wet markets" which sell fresh meat and seafood from a variety of animals including bats and pangolins. The pneumonia was found to be caused by a virus identified as "severe acute respiratory syndrome coronavirus 2" (SARS-CoV-2), with the associated disease subsequently termed coronavirus disease 2019 (COVID-19) by the World Health Organization (WHO) [1,2]. Genomic analysis showed that COVID-19 is phylogenetically related to SARS-like bat viruses. Hence, bats could be the possible source of the viral replication [1]. Pangolins have also been identified as a potential intermediate host of COVID-19 [3]. This newly identified virus is highly transmittable and pathogenically different from SARS-CoV, MERS-CoV, avian influenza, influenza, and other common respiratory viruses. Concerning the outbreak of COVID-19, on January 30, 2020, the WHO declared the outbreak of the novel Coronavirus disease as a Public Health Emergency of International Concern (PHEIC) [4]. The rapid worldwide spread of disease resulted in a global pandemic declaration on March 11, 2020. Clinical symptoms of patients infected with COVID-19 are similar to other viral upper respiratory diseases such as Influenza, respiratory syncytial virus (RSV), and bacterial pneumonia. The most common presenting symptoms experienced by patients include dry cough, sore throat, fever, dyspnea, diarrhea, myalgia, shortness of breath and bilateral lung infiltrates, observable on clinical imaging such as chest X-ray. Other symptoms are headache, vomiting, pleurisy, sneezing, rhinorrhea, and nasal congestion. Patients with more severe COVID-19 have developed critical complications, including septic shock, pulmonary edema, cardiac injury, acute kidney injury, Acute Respiratory Distress Syndrome (ARDS) and even Multi-Organ Failure (MOF) [4,5]. At present, there is no clinically approved antiviral drug or vaccine available to treat COVID-19. The reproduction number (R0), defined as the expected number of susceptible cases directly generated by one infectious case of COVID-19 infection, is estimated to 3.77 [6,7]. Despite global efforts of travel restrictions and quarantine, while the epidemic continues to decline in China, the incidence of novel COVID-19 continues to rise globally, with over 1.6 million confirmed cases and over 100,000 deaths worldwide, at the time of this writing [8]. As of April 2020, substantial new incidence of COVID-19 cases have been reported in 211 countries with significant confirmed cases in South Korea, Italy, Iran, Japan, Germany, and France [9]. The early spread of new COVID-19 cases was associated with recent travel to China; however, community spread is now common globally. The greatest number of new cases occur through close contact human-to–human transmission (approximately 6 feet) by respiratory droplets [5]. Contamination also can occur through infected surfaces with subsequent contact with the eyes, nose, or mouth.

The genetic characteristics of the Coronavirus should be well understood to fight against this virus. Coronavirus is a single-stranded RNA virus consisting of approximately 27–32 kb with particle size ranged from 65–125 nm in diameter [1]. An illustration of COVID-19 is shown in Fig. 1. A transmission electron microscopic image of a case of COVID-19 is also demonstrated in Fig. 2.

In light of this, it is evident that early detection of COVID-19 is necessary to interrupt the spread of COVID-19 and prevent transmission by early isolation of patients, trace and quarantine of close contacts. In patients with COVID-19, accurate monitoring of the disease progression is a critical component of disease management. While not recommended for primary diagnosis of COVID-19 in Canada, medical imaging modalities such as chest X-ray and Computed Tomography (CT) play an important role in confirming diagnosis of positive COVID-19 pneumonia as well as monitoring the progression of the disease. These types of images show an extent of irregular ground-glass opacities that progress rapidly after COVID-19 symptom onset. These abnormalities peaked during days 6–11 of the illness. The second most predominant pattern of lung opacity abnormalities peak during days 12–17 of the illness [12]. Computer-Aided Diagnosis (CAD) systems that incorporate X-ray and CT image processing techniques and deep learning algorithms could assist physicians as diagnostic aides for COVID-19 and help provide a better understanding of the progression the disease [13].

1.1. Related research

Hemdan et al. [14] developed a deep learning framework, COVIDX-Net, to diagnose COVID-19 in X-ray Images. A comparative study of different deep learning architectures including VGG19, DenseNet201, ResNetV2, InceptionV3, InceptionResNetV2, Xception and MobileNetV2 is provided by authors. The public dataset of X-ray images was provided by Dr. Joseph Cohen [15] and Dr. Adrian Rosebrock [16]. The provided dataset included 50 X-ray images, divided into two

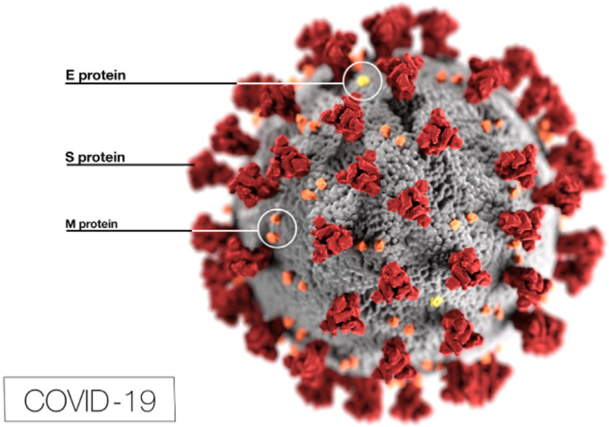

Fig. 1 – The illustration of COVID-19, created at the Centers for Disease Control and Prevention (CDC) [10]. The protein particles E, S, and M are located on the outer surface of the. virus particle.

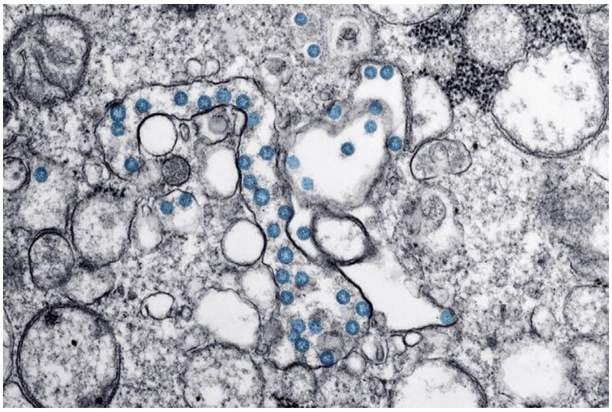

Fig. 2 – Transmission electron microscopic image of a case of COVID-19. The spherical viral particles, colorized blue, contain cross-sections through the viral genome, seen as black dots [11].

classes as 25 normal cases and 25 positive COVID-19 images. Hemdan's results demonstrated VGG19 and DenseNet201 models achieved the best performance scores among counterparts with 90.00% accuracy.

Barstugan et al. [17] proposed a machine learning approach for COVID-19 classification from CT images. Patches with different sizes 16 × 16, 32 × 32, 48 × 48, 64 × 64 were extracted from 150 CT images. Different hand-crafted features such as Grey Level Co-occurrence Matrix (GLCM), Local Directional Pattern (LDP), Grey Level Run Length Matrix (GLRLM), Grey-Level Size Zone Matrix (GLSZM), and Discrete Wavelet Transform (DWT) algorithms were employed. The extracted features were fed into a Support Vector Machine (SVM) [18] classifier on 2-fold, 5-fold and 10-fold cross-validations. The best accuracy of 98.77% was obtained by GLSZM feature extractor with 10-fold cross-validation.

Wang and Wong [19] designed a tailored deep learning-based framework, COVID-Net, developed for COVID-19 detection from chest X-ray images. The COVID-Net architecture was constructed of combination of 1×1 convolutions, depth-wise convolution and the residual modules to enable design deeper architecture and avoid the gradient vanishing problem. The provided dataset consisted of a combination of COVID chest X-ray dataset provided by Dr. Joseph Cohen [15], and Kaggle chest X-ray images dataset [20] for a multi-class classification of normal, bacterial infection, viral infection (non-COVID) and COVID-19 infection. Obtained accuracy of this study was 83.5%.

In a study conducted by Maghdid et al. [21], a deep learning-based method and transfer learning strategy were used for automatic diagnosis of COVID-19 pneumonia. The proposed architecture is a combination of a simple convolutional neural network (CNN) architecture (one convolutional layer with 16 filters followed by batch normalization, rectified linear unit (ReLU), two fully-connected layers) and a modified AlexNet [22] architecture with the feasibility of transfer learning. The proposed modified architecture achieved an accuracy of 94.00%.

Ghoshal and Tucker [23] investigated the diagnostic uncertainty and interpretability of deep learning-based methods for COVID-19 detection in X-ray images. Dropweights based Bayesian Convolutional Neural Networks (BCNN) were used to estimate uncertainty in deep learning solutions and provide a level of confidence of a computer-based diagnosis for a trusted clinician setting. To measure the relationship between accuracy and uncertainty, 70 posterioranterior (PA) lung X-ray images of COVID-19 positive patients from the public dataset provided by Dr. Joseph Cohen [15] were selected and balanced by Kaggle's Chest X-ray Images dataset [20]. To prepare the dataset, all images were resized to 512 × 512 pixels. A transfer learning strategy and real-time data augmentation strategies were employed to overcome the limited size of the dataset. The proposed Bayesian inference approach obtained the detection accuracy of 92.86% on X-ray images using VGG16 deep learning model.

Hall et al. [24] used a VGG16 architecture and transfer learning strategy with 10-fold cross-validation trained on the dataset from Dr. Joseph Cohen [15]. All images were rescaled to 224 × 224 pixels and a data augmentation strategy was employed to increase the size of dataset. The proposed approach achieved an overall accuracy 96.1% and overall Area Under Curve (AUC) of 99.70% on the provided dataset.

Farooq and Hafeez [25] proposed a fine-tuned and pre-trained ResNet-50 architecture, COVID-ResNet, for COVID-19 pneumonia screening. To improve the generalization of the training model, different data augmentation methods including vertical flip, random rotation (with angle of 15 degree), along with the model regularization were used. The proposed method achieved the accuracy of 96.23% on a multi-class classification of normal, bacterial infection, viral infection (non-COVID-19) and COVID-19 infection dataset.

1.2. *Motivation and contributions*

The main motivation of this study is to present a generic feature extraction method using convolutional neural networks that does not require handcrafted or very complex features from input data while being easily applied to different modalities such as X-ray and CT images. Another primary goal is to reduce the generalization error while achieving a more accurate diagnosis. The contributions are summarized as follows:

- Deep convolutional feature representation [26–28] is used to extract highly representative features using state-of-the-art deep CNN descriptors. The employed approach is able to discriminate between COVID-19 and healthy subjects from chest X-ray and CT images and hence produce higher accuracy in comparison to other works presented in the literature. To the best of our knowledge, this research is the first comprehensive study of the application of machine learning (ML) algorithms (15 deep CNN visual feature extractor and 6 ML classifier) for automatic diagnoses of COVID-19 from X-ray and CT images.

- To overcome the issue of over-fitting in deep learning due to the limited number of training images, a transfer-learning strategy is adopted as the training of very deep CNN models from scratch requires a large number of training data.

- No data augmentation or extensive pre-processing methods are applied to the dataset in order to increase the generalization ability and also reduce bias toward the model performance.

- The proposed approach reduces the detection time dramatically while achieving satisfactory accuracy, which is a superior advantage for developing real or near real-time inferences on clinical applications.

- With extensive experiments, we show that the combination of a deep CNN with Bagging trees classifier achieves very good classification performance applied on COVID-19 data despite the limited number of image samples.

- Finally, we developed an end to end web-based detection system to simulate a virtual clinical pipeline and facilitate the screening of suspicious cases.

The rest of this paper is organized as follows. The proposed methodology for automatically classifying COVID-19 and healthy cases is explained in Section 2. The dataset description, experimental settings and performance metrics are given in Section 3. A brief discussion and results analysis are provided in Section 4, and finally, the conclusion is presented in Section 5.

2. Proposed methodology

Few studies have been published on the application of deep CNN feature descriptors to X-ray and CT images. Each of the CNN architectures is constructed by different modules and convolution layers that aid in extracting fundamental and prominent features from a given input image. Briefly, in the first step, we collect available public chest X-ray and CT images. In the next step, we pre-processed the provided dataset using standard image normalization techniques to improve the quality of visual information of the input data. Once input images are prepared, we fed them into the feature extraction phase with the state-of-the-art CNN descriptors to extract deep features from each input image. For the training phase, the generated features are then fed into machine learning classifiers such as Decision Tree (DT) [29], Random Forest (RF) [30], XGBoost [31], AdaBoost [32], Bagging classifier [33] and LightGBM [34]. Finally, a 10-fold cross-validation technique was adopted to evaluate the average generalization performance of the classifiers in each experiment on test images.

2.1. *Feature extraction using transfer learning*

The concept of transfer learning has been introduced for solving deep learning problems arising from insufficiently labeled data, or when the CNN model is too deep and complex. Aiming to tackle these challenges, studies in a variety computer vision tasks demonstrated the advantages of transfer learning strategies from an auxiliary domain in improving the detection rate and performance of a classifier [35–37]. In a transfer learning strategy, we transfer the weights already learned on a cross-domain dataset into the current deep learning task instead of training a model from scratch. With the transfer learning strategy, the deep CNN can obtain general features from the source dataset that cannot be learned due to the limited size of the dataset in the current task [38]. Transfer learning strategies have various advantages, such as avoiding the overfitting issue when the number of training samples is limited, reducing the computational resources, and also speeding up the convergence of the network [39,40].

2.2. *CNN descriptor*

Effective feature extraction is one of the most important steps toward learning rich and informative representations from raw input data to provide accurate and robust results. The small or imbalanced size of the training samples poses a significant challenge for the training of a deep CNN where data dimensionality is much larger than the number of samples leading to over-fitting. Although various strategies, e.g. data augmentation [41], transfer learning [42] and fine-tuning [43], may reduce the problem of insufficient or imbalance training data, the detection rate of the CNN model may degrade due to the over-fitting issue. Since the overall performance obtained by a fine-tuning method in the initial experiments for this study was not significant, we employed a different approach inspired by [26–28] known as deep convolutional feature representation. In this method, we used pre-trained well-established CNN models as a visual feature extractor to encode the input images into a feature vector of sparse descriptors of low dimensionality. Then the computed encoded feature vectors produced by CNN architectures are fed into different classifiers, i.e. machine learning algorithms, to yield the final prediction. This lower dimension vector significantly reduces the risk of over-fitting and also the training time. Different robust CNN architectures such as MobileNet, DenseNet, Xception, InceptionV3, InceptionResNetV2, ResNet, VGGNet, NASNet are selected for feature extraction. These architectures are selected based on their (i) satisfying performance in real-world medical setting (ii) adaptation towards real-time (or near real-time) image diagnosis support system and, (iii) feasibility of transfer learning different computer vision tasks detection, segmentation, classification or visual attributes. tasks [44–47]. Fig. 3. illustrates the visual features extracted by VGGNet architecture from an X-ray image of a COVID-19 positive patient.

2.2.1. *MobileNet*

MobileNet architecture [48] was designed by Google research team for object recognition on mobile devices. This architecture introduced the depth-wise separable convolution and 1×1 point-wise convolution layers. MobileNet architecture achieved an accuracy as the same level as VGG16 with 32 times less parameters than standard convolutions and won

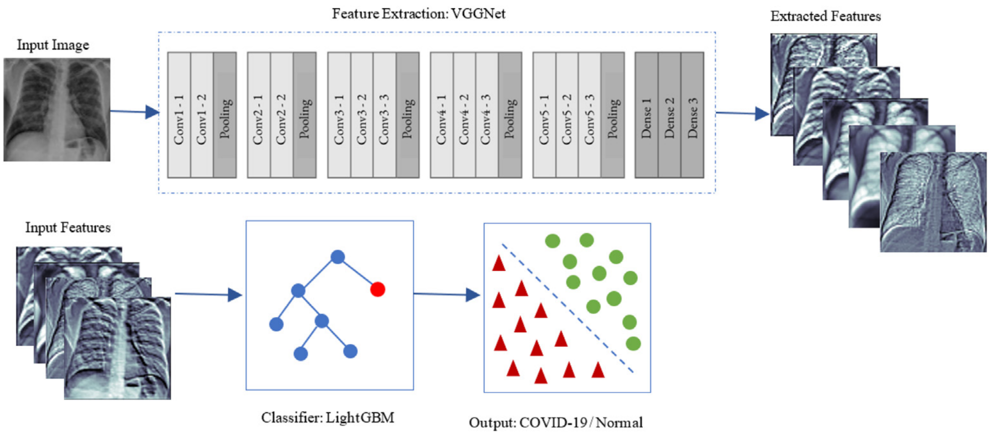

Fig. 3 – General framework of the proposed method with VGGNet as feature extractor.

ImageNet dataset while is 27 times less computationally intensive. Depth-wise convolution applies single spatial filter to each input channel. Point-wise convolution is a standard convolution with the kernel size of (1 × 1) that computes a linear combination of different input channels. This operation can significantly reduce the dimensionality of the feature maps.

2.2.2. *Densely connected convolutional networks*

Densely connected convolutional networks (DenseNet) architecture proposed by Huang et al. [49] is an extension for ResNet architecture. Summation operation is used to connect layers to each other in this architecture. The summation operation helps further improvement of the generalization ability and addressing the problem of vanishing gradient better in compare to ResNet architecture. In this approach, the features extracted from each layer are reused as input for the next layers. The idea of reusing feature maps could help further improvement of the final performance.

2.2.3. *Xception*

The Xception architecture, introduced by François Chollet [50] in 2017, is an extension of the Inception architecture and stands for extreme inception. Inception modules in this architecture are replaced with depthwise separable convolutions and residual connections. The depthwise separable convolution could decrease the computational cost and memory requirements. Xception architecture contains 14 modules with 36 convolutional layers. Linear residual connections are used to connect all modules together, except for the first and last module.

2.2.4. *InceptionV3*

InceptionV3 architecture proposed by Szegedy et al. [51] in 2014. InceptionV3 architecture introduced the concept of inception module and won the ImageNet competition. This architecture consists of 159 layers in total and is the third generation of Inception model. In Inception module instead of using one type of kernel size (i.e. 3 × 3, or 5 × 5), different convolution size, i.e. 1 × 1, 3 × 3, and 5 × 5 filter size is used. The main idea of employing different convolution size allows to extract multi-level features from the input image in every convolution operation. In this architecture also pointwise 1 × 1 convolution is used to reduce the number of parameters. The pointwise convolution helps reduce the computational cost.

2.2.5. *InceptionResNetV2*

Both Inception modules and Residual blocks are combined in InceptionResNetV2 [52]. The combination of these modules helps achieve a better performance with relatively low computational cost.

2.2.6. *Deep residual learning network*

Deep Residual Learning Network (ResNet) proposed by He et al. [53]. This architecture won ILSVRC classification task with good results on ImageNet and MS-COCO object detection competitions. This architecture introduced the concept of residual blocks. The main goal of residual blocks is to add a connection (instead of concatenation) from the input of the first block to the output of the next block in order to train a more deeper network with better recognition ability. This architecture can solve the issues of vanishing gradients and parameter explosion by shortcut connection using the residual blocks.

2.2.7. *VGG-Net*

Visual Geometry Group (VGG-Net) [54] was proposed by Karen Simonyan and Andrew Zisserman. This architecture obtained top performances on ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014. This architecture provides better features extraction from input images by using 3 × 3 filter size. VGG16 and VGG19 are two versions of VGG-Net architecture with different depths and layers.

2.2.8. *Neural architecture search net*

Neural Architecture Search Net (NAS-Net) [55] architecture was proposed by Google Brain in 2017 and achieved the state-of-the-art results on the CIFAR10 dataset. This architecture search for the best convolutional layer using recurrent neural networks on the CIFAR dataset. Then the selected layers are transferred to the ImageNet dataset. Next the selected convolutional layer are stacked together to produce the final architecture. A new regularization method called ScheduledDropPath is introduced in this architecture to further improve the generalization ability. However, the computational cost of NAS-Net architecture is expensive.

For features extraction stage, we employ standard pre-trained networks from Keras distribution. The dimension of the output vector generated by each CNN architecture is presented in the Table 1.

3. Experiments

3.1. Dataset description

In order to evaluate the performance of our feature extracting and classifying approach, we used the public dataset of X-ray images provided by Dr. Joseph Cohen available from a GitHub repository [15]. We used the available 117 chest X-ray images and 20 CT images (137 images in total) of COVID-19 positive cases. We also included 117 images of healthy cases of X-ray images from Kaggle Chest X-ray Images (Pneumonia) dataset available at [20] and 20 images of healthy cases of CT images from Kaggle RSNA Pneumonia Detection dataset available at [56] to balance the dataset with both positive and normal cases. Fig. 4 shows examples of confirmed COVID-19 images extracted from the provided dataset. The X-ray images of confirmed COVID-19 infection demonstrate different shapes of "pure ground glass" also known as hazy lung opacity with irregular linear opacity depending the disease progress [12].

3.2. Data pre-processing

The images within the dataset were collected from multiple imaging clinics with different equipment and image acquisition parameters; therefore, considerable variations exist in images' intensity. The proposed method in this study avoids extensive pre-processing steps to improve the generalization ability of the CNN architecture. This helps to make the model more robust to noise, artifacts and variations in input images during feature extraction phase. Hence, we only employed two standard pre-processing steps in training deep learning models to optimize the training process.

Table 1 – Feature vector of each deep CNN architecture.

CNN feature extractor	Output feature vector
Xception	2048
VGG19	512
VGG16	512
ResNet50	2048
ResNet152	2048
ResNet101V2	2048
ResNet50V2	2048
ResNet152V2	2048
NASNetMobile	1056
NASNetLarge	4032
MobileNet	1024
InceptionV3	2048
InceptionResNetV2	1536
DenseNet201	1920
DenseNet121	1024

- Resizing: The images in this dataset vary in resolution and dimension, ranging from 365 × 465 to 1125 × 859 pixels; therefore, we re-scaled all images of the original size to the size of 600 × 450 pixels to obtain a consistent dimension for all input images. The input images were also separately resized to 331 × 331 pixels and 224 × 224 pixels as required for NASNetLarge and NASNetMobile architectures, respectively.

- Image normalization: For image normalization, first, we re-scaled the intensity values of the pixels using ImageNet mean subtraction as a pre-processing step. The ImageNet mean is a pre-computed constant derived from the ImageNet database [22]. Another essential pre-process step is intensity normalization. To accomplish this, we normalized the intensity values of all images from [0, 255] to the standard normal distribution by min–max normalization to the intensity range of [0, 1], which is computed as:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

where x is the pixel intensity. x_{min} and x_{max} are minimum and maximum intensity values of the input image in Eq. 1. This operation helps to speed up the convergence of the model by removing the bias from the features and achieve a uniform distribution across the dataset.

3.3. Experimental setting

The configurations of hyperparameters of each machine learning model (LightGBM, bagging, Adaboost, random forest, XGBoost, and decision tree) is highlighted in Table 2.

3.4. Evaluation criteria

To measure the prediction performance of the methods in this study, we utilized common evaluation metrics such as recall, precision, accuracy and f1-score. According to Eqs. (2)–(5) True positive (TP) is the number of instances that correctly predicted; false negative (FN) is the number of instances that incorrectly predicted. True negative (TN) is the number of negative instances that predicted correctly, while false positive (FP) is the number of negative instances incorrectly predicted. Given TP, TN, FP and FN, all evaluation metrics were calculated as follows:

Recall or sensitivity is the measure of COVID-19 cases that are correctly classified. Recall is critical, especially in the medical field and is given by:

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

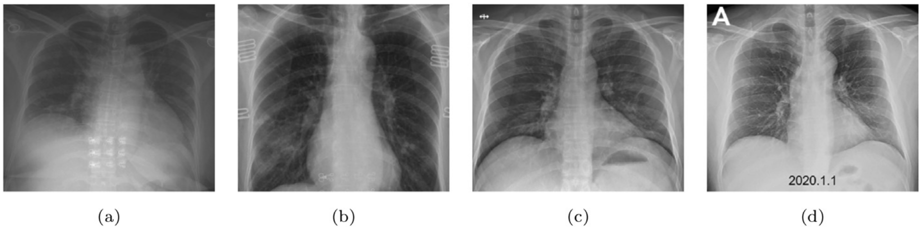

Fig. 4 – Chest X-ray images of four confirmed COVID-19 pneumonia: (a) 52-year old female, presenting diffuse infiltrates in the bilateral lower lungs; (b) 59-year old female, demonstrating right infahilar airspace opacities; (c) 35-year old male, presenting stable streaky opacities in the lung bases, indicating likely atypical pneumonia; the opacities have steadily increased in density over time; (d) 42-year old male, presenting opacities in the left lower and right upper lobes on day 7 after the onset of symptoms.

Table 2 – Hyper-parameters considered in this study.

Methods	Parameters
LightGBM	n_estimators: (Number of trees in the forest): 1000 Class_wight: Balanced, Reg_alpha: 0.1, Reg_lambda: 0.1, Learning rate: 0.001, Num_leaves: 400, Boosting: "dart"
Bagging	DecisionTree Classifier, n_estimators: (Number of trees in the forest): 5
Adaboost	DecisionTree Classifier, n_estimators: (Number of trees in the forest): 300
Random Forest	n_estimators: (Number of trees in the forest): 5
XGBoost	Default setting
Decision Tree	Default setting

Precision or positive predictive value is defined as the percentage of correctly classified labels in truly positive patients and is given as:

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Accuracy shows the number of correctly classified cases divided by the total number of test images, and is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

F1-score, also known as F-measure, is defined as the weighted average of precision and recall that combines both the precision and recall together. F-measure is expressed as:

$$\mathrm{F1-score} = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (5)$$

4. Experimental methods and results

Diagnostic imaging modalities, such as chest radiography and CT are playing an important role in confirming the primary diagnosis from the Polymerase Chain Reaction (PCR) test for COVID-19. Medical imaging is also playing a critical in monitoring the progression of the disease and patient care. Extracting features from radiology modalities is an essential step in training machine learning models since the model performance directly depends on the quality of extracted features.

Fig. 5 shows some examples of output feature vector generated by the first convolution layer of VGG16 architecture. This figure displays the extracted features for a COVID-19 case (left column) and a healthy case (right column) cases.

Motivated by the success of deep learning models in computer vision [57], the focus of this research is to provide an extensive comprehensive study on the classification of COVID-19 pneumonia in chest X-ray and CT imaging using features extracted by the state-of-the-art deep CNN architectures and trained on machine learning algorithms. The 10-fold cross-validation technique was adopted to evaluate the average generalization performance of the classifiers in each experiment. For all CNNs, the network weights were initialized from the weights trained on ImageNet. The Windows based computer system used for this work had an Intel(R) Core(TM) i7-8700 K 3.7 GHz processors with 32 GB RAM. The training and testing process of the proposed architecture for this experiment was implemented in Python using Keras package with Tensorflow backend as the deep learning framework backend and run on Nvidia GeForce GTX 1080 Ti GPU with 11 GB RAM.

Table 3 and Fig. 6 summarize the accuracy performance of six machine learning algorithms, namely, DT, RF, XGBoost, AdaBoost, Bagging classifier and LightGBM on the feature extracted by deep CNNs. Each entry in Table 3, is in the format ($\mu \pm \sigma$) where μ is the average classification accuracy and σ is standard deviation. Analyzing Table 3 the topmost result

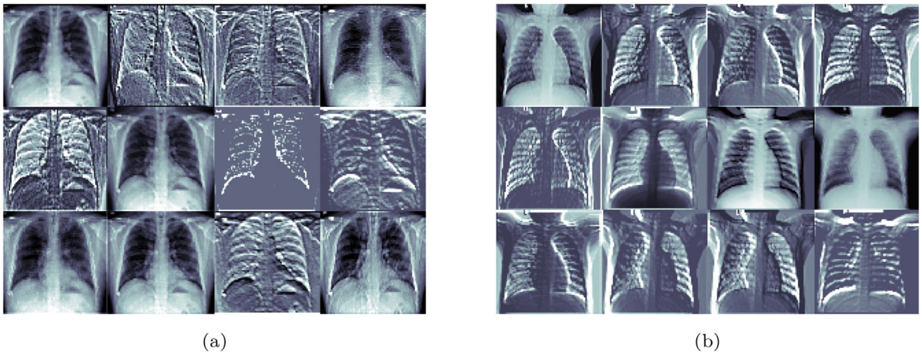

Fig. 5 – Deep features learned by VGG16 model: (a) COVID-19 case; (b) healthy case.

Table 3 – Comparison of classification performance ($\mu \pm \sigma$) of different machine learning models measured by accuracy. The bold value indicates the best result; underlined value represents the second-best result of the respective category.

	Decision Tree	Random Forest	XGBoost	AdaBoost	Bagging	LightGBM
MobileNet	83.00 ± 0.26	93.00 ± 0.23	95.00 ± 0.16	80.00 ± 0.17	96.00 ± 0.11	82.00 ± 0.28
DenseNet121	92.00 ± 0.15	90.00 ± 0.21	94.00 ± 0.16	92.00 ± 0.19	**99.00 ± 0.07**	96.00 ± 0.11
DenseNet201	84.00 ± 0.26	90.00 ± 0.24	90.00 ± 0.18	87.00 ± 0.25	96.00 ± 0.11	87.00 ± 0.17
Xception	95.00 ± 0.17	90.00 ± 0.19	96.00 ± 0.11	93.00 ± 0.20	96.00 ± 0.11	96.00 ± 0.11
InceptionV3	82.00 ± 0.22	84.00 ± 0.29	88.00 ± 0.15	80.00 ± 0.12	95.00 ± 0.12	84.00 ± 0.16
InceptionResNetV2	84.00 ± 0.31	93.00 ± 0.16	93.00 ± 0.19	87.00 ± 0.33	94.00 ± 0.12	88.00 ± 0.21
ResNet50	89.00 ± 0.17	90.00 ± 0.15	93.00 ± 0.16	94.00 ± 0.12	93.00 ± 0.16	98.00 ± 0.09
ResNet152	93.00 ± 0.12	92.00 ± 0.16	93.00 ± 0.16	94.00 ± 0.17	91.00 ± 0.22	93.00 ± 0.20
VGG16	90.00 ± 0.19	91.00 ± 0.19	88.00 ± 0.19	90.00 ± 0.19	90.00 ± 0.19	85.00 ± 0.19
VGG19	90.00 ± 0.19	87.00 ± 0.21	88.00 ± 0.19	90.00 ± 0.19	90.00 ± 0.19	85.00 ± 0.25
NASNetLarge	82.00 ± 0.23	88.00 ± 0.19	89.00 ± 0.17	81.00 ± 0.23	93.00 ± 0.19	82.00 ± 0.26
NASNetMobile	87.00 ± 0.17	88.00 ± 0.22	94.00 ± 0.19	87.00 ± 0.17	93.00 ± 0.19	89.00 ± 0.17
ResNet50V2	87.00 ± 0.12	96.00 ± 0.11	92.00 ± 0.19	90.00 ± 0.18	95.00 ± 0.12	88.00 ± 0.10
ResNet101V2	79.00 ± 0.32	89.00 ± 0.24	89.00 ± 0.28	76.00 ± 0.32	95.00 ± 0.12	78.00 ± 0.26
ResNet152V2	90.00 ± 0.27	86.00 ± 0.26	93.00 ± 0.16	89.00 ± 0.20	96.00 ± 0.11	88.00 ± 0.28

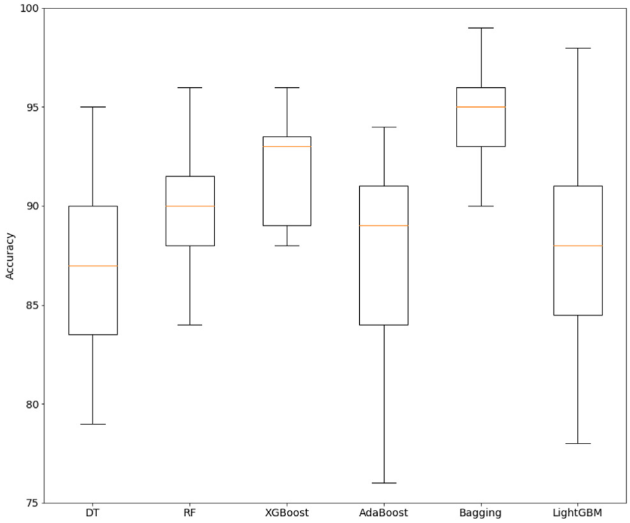

Fig. 6 – Performance of different ML classifiers on the COVID-19 pneumonia classification.

was obtained by Bagging classifier with a maximum of 99.00% ± 0.09 accuracy on features extracted by DenseNet121 architecture (with feature extraction time of 9.306 s and training time of 30.748 s in Table 7), which is the highest result reported in the literature for COVID-19 classification of this dataset. It is also inferred from Table 3 that the second-best result obtained by ResNet50 feature extractor and LightGBM classifier (with feature extraction time of 10.206 s and training time of 0.960 s in Table 7) with an overall accuracy of 98.00 ± 0.09. Comparing the first and second winners among all combinations, the classification accuracy of DenseNet121 with Bagging is slightly better (1%) than ResNet50 with LightGBM, while the training time of the second winner is tempting, almost 30 times better than the first winner in terms of accuracy. Although Bagging is a slow learner, it has the lowest standard deviation and hence is more stable than other learners.

The results also demonstrate that the detection rate is worst on the features extracted by ResNet101V2 trained by the AdaBoost classifier with 76.00 ± 0.32 accuracy. Figs. 6 and 7 demonstrate box-plot distributions of deep CNNs feature extractors and classification accuracy from the 10-fold cross-validation. Circles in Fig. 7 represent outliers. In Tables 4–6, the obtained precision, recall, and F1-score of the features extracted by deep CNN architectures trained by different learners are presented respectively. As given in these tables, the highest precision, recall, and F1-score rates are

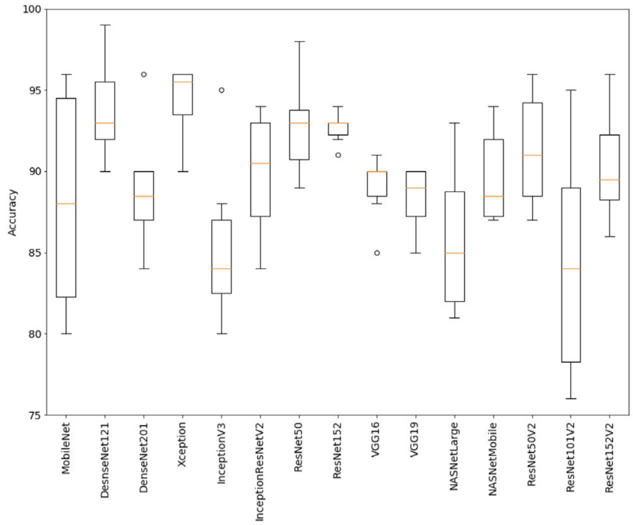

Fig. 7 – Performance of the deep CNNs feature extractors and Bagging classifier on the COVID-19 pneumonia classification.

achieved by MobileNet and InceptionV3 feature vector trained on Bagging tree classifier with value of 99.00% precision, recall, and F-score. The XGBoost and Bagging classifiers also yielded the second-best results with values of (98.00, 98.00, 98,00)% precision, recall, and F-score rates with features extracted by DenseNet121, DenseNet201 and Xception architectures. Similar conclusions can be drawn for other models. The experimental results indicate that the performance of the deep CNNs using DenseNet121, DenseNet201, MobileNet, Xception and InceptionV3 models trained by Bagging tree and XGBoost classifiers yield satisfactory results and outperforms other state-of-the-art CNNs and learners in COVID-19 classification. Based on the obtained results, we believe that by discarding the irrelevant features using sparse descriptors of low dimensionality features extracted by deep CNN models instead of training a deep CNNs model can be considered as a successful improvement of the performance of a machine learning algorithms. Our obtained results agree with the top-performing ML classifiers of Bagging and LightGBM. The best pre-trained visual feature extractor so far was DenseNet121, MobileNet and InceptionV3 rather than counterpart architectures for COVID-19 image classification.

Although the approach presented here shows satisfying performance, it also has limitations classifying more challenging instances with vague, low contrast boundaries, and the presence of artifacts. Some examples of these cases are illustrated in Fig. 9. Finally, comparison of the feature extraction time using deep CNN models and training with ML algorithms are shown in Table 7 and Fig. 8. The extraction time of the DenseNet201 architectures on the total of 274 images was computed with 38.227 s (about 0.13 s per image) was the longest visual feature extractor and NASNetMobile was the fastest visual feature extractor by 7.786 s (about 0.028 s per image). DenseNet121 architecture as the best model took 9.306 s (about 0.03 s per image) for feature extraction phase and 30.748 s (about 0.11 s per image) for the training phase on Bagging tree classifier. ResNet50 architecture as the second-best visual feature extractor took 10.206 s (about 0.03 s per image) for feature extraction phase and the training time of 0.960 s (about 0.003 s per image) on LightGBM classifier. In conclusion, the extraction and training time of the proposed approach is considerate significantly low in comparison with training a deep CNN model from scratch which implies faster computation time and lower resource consumption.

The high-level feature vectors of the last fully connected of the best combination of features and learner (DenseNet121 and Bagging classifier) are extracted and trained from the test dataset. We use the feature extractor model (DenseNet121) to extract convolution features. Then, we convert features extracted from each original image into a row vector. Referring to the Table 1, the dimension of the row vector is 1024. We sort all deep feature representations and their correspond labels, accordingly. After that, we adopt the T-SNE model to visualize the classification results from the bagging classifier. The visualization result is shown in Fig. 10 with two different colors represent different classes of each test data (dark blue

Table 4 – Comparison of classification precision metric of different machine learning models. The bold value indicates the best result; underlined value represents the second-best result of the respective category.

	Decision Tree	Random Forest	XGBoost	AdaBoost	Bagging Classifier	LightGBM
MobileNet	89.00%	88.00%	93.00%	85.00%	**99.00%**	90.00%
DenseNet121	96.00%	97.00%	98.00%	95.00%	96.00%	95.00%
DenseNet201	94.00%	94.00%	95.00%	94.00%	98.00%	94.00%
Xception	92.00%	95.00%	90.00%	89.00%	98.00%	93.00%
InceptionV3	85.00%	85.00%	96.00%	85.00%	**99.00%**	82.00%
InceptionResNetV2	88.00%	96.00%	95.00%	90.00%	95.00%	93.00%
ResNet50	95.00%	89.00%	94.00%	96.00%	95.00%	94.00%
ResNet152	90.00%	91.00%	95.00%	91.00%	93.00%	89.00%
VGG16	94.00%	93.00%	94.00%	89.00%	92.00%	89.00%
VGG19	94.00%	93.00%	94.00%	89.00%	92.00%	89.00%
NASNetLarge	89.00%	91.00%	94.00%	90.00%	95.00%	91.00%
NASNetMobile	89.00%	87.00%	95.00%	88.00%	93.00%	88.00%
ResNet50V2	92.00%	89.00%	94.00%	88.00%	96.00%	91.00%
ResNet101V2	87.00%	89.00%	94.00%	86.00%	96.00%	78.00%
ResNet152V2	91.00%	94.00%	96.00%	91.00%	97.00%	91.00%

Table 5 – Comparison of classification recall metric of different machine learning models. The bold value indicates the best result; underlined value represents the second-best result of the respective category.

	Decision Tree	Random Forest	XGBoost	AdaBoost	Bagging Classifier	LightGBM
MobileNet	89.00%	88.00%	93.00%	84.00%	**99.00%**	90.00%
DenseNet121	96.00%	96.00%	98.00%	95.00%	96.00%	95.00%
DenseNet201	94.00%	94.00%	95.00%	94.00%	98.00%	94.00%
Xception	90.00%	95.00%	90.00%	89.00%	98.00%	93.00%
InceptionV3	85.00%	85.00%	96.00%	85.00%	**99.00%**	82.00%
InceptionResNetV2	88.00%	95.00%	95.00%	90.00%	95.00%	93.00%
ResNet50	95.00%	89.00%	94.00%	96.00%	95.00%	94.00%
ResNet152	89.00%	91.00%	95.00%	90.00%	93.00%	89.00%
VGG16	94.00%	93.00%	94.00%	89.00%	91.00%	89.00%
VGG19	94.00%	93.00%	94.00%	89.00%	91.00%	89.00%
NASNetLarge	88.00%	91.00%	94.00%	90.00%	95.00%	90.00%
NASNetMobile	89.00%	87.00%	95.00%	88.00%	93.00%	85.00%
ResNet50V2	91.00%	89.00%	94.00%	88.00%	96.00%	91.00%
ResNet101V2	87.00%	88.00%	94.00%	85.00%	96.00%	77.00%
ResNet152V2	90.00%	94.00%	96.00%	90.00%	96.00%	91.00%

Table 6 – Comparison of classification f1-score metric of different machine learning models. The bold value indicates the best result; underlined value represents the second-best result of the respective category.

	Decision Tree	Random Forest	XGBoost	AdaBoost	Bagging Classifier	LightGBM
MobileNet	89.00%	88.00%	93.00%	84.00%	**99.00%**	91.00%
DenseNet121	96.00%	96.00%	98.00%	95.00%	96.00%	95.00%
DenseNet201	94.00%	94.00%	95.00%	94.00%	98.00%	94.00%
Xception	90.00%	95.00%	90.00%	89.00%	98.00%	93.00%
InceptionV3	85.00%	85.00%	96.00%	85.00%	**99.00%**	82.00%
InceptionResNetV2	88.00%	95.00%	95.00%	90.00%	95.00%	93.00%
ResNet50	95.00%	89.00%	94.00%	96.00%	95.00%	94.00%
ResNet152	89.00%	91.00%	95.00%	90.00%	93.00%	89.00%
VGG16	94.00%	93.00%	94.00%	89.00%	91.00%	89.00%
VGG19	94.00%	93.00%	94.00%	89.00%	91.00%	89.00%
NASNetLarge	88.00%	91.00%	94.00%	90.00%	95.00%	90.00%
NASNetMobile	89.00%	87.00%	95.00%	88.00%	93.00%	85.00%
ResNet50V2	91.00%	89.00%	94.00%	88.00%	96.00%	91.00%
ResNet101V2	87.00%	88.00%	94.00%	85.00%	96.00%	77.00%
ResNet152V2	90.00%	94.00%	96.00%	90.00%	96.00%	91.00%

Table 7 – The time for feature extraction of deep CNN models and training on ML algorithms using Intel(R) Core (TM) i7-8700 K 3.7 GHz processors with 32 GB RAM, Nvidia GeForce GTX 1080 Ti GPU with 11 GB RAM.

	Extraction Time (s)	DT (s)	RF (s)	XGBoost (s)	AdaBoost (s)	Bagging Classifier (s)	LightGBM (s)
MobileNet	8.803	0.022	0.008	0.438	0.023	33.535	1.097
DenseNet121	9.306	0.017	0.009	0.362	0.021	30.748	0.897
DenseNet201	38.227	0.035	0.009	0.684	0.034	33.446	1.573
Xception	10.819	0.042	0.009	0.787	0.044	35.144	1.612
InceptionV3	11.825	0.045	0.009	0.86	0.048	37.54	1.98
InceptionResNetV2	14.151	0.035	0.009	0.575	0.035	33.562	1.169
ResNet50	10.206	0.034	0.009	0.694	0.04	33.232	0.96
ResNet152	15.769	0.031	0.01	0.653	0.031	32.347	1.114
VGG16	14.746	0.009	0.008	0.2	0.012	29.51	0.498
VGG19	14.359	0.01	0.008	0.2	0.013	29.336	0.494
NASNetLarge	13.131	0.066	0.01	1.409	0.067	38.337	2.542
NASNetMobile	7.786	0.024	0.009	0.429	0.024	32.782	0.93
ResNet50V2	10.204	0.044	0.009	0.691	0.045	34.369	1.798
ResNet101V2	12.435	0.047	0.009	0.776	0.048	0.9634	1.577
ResNet152V2	16.67	0.031	0.009	0.73	0.032	34.56	1.514

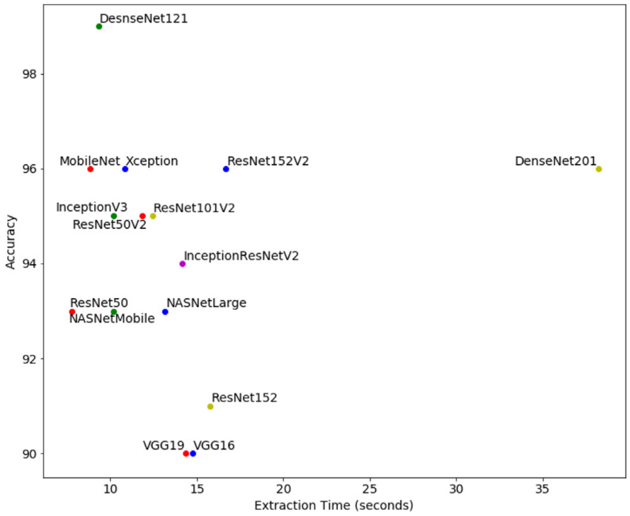

Fig. 8 – Feature extraction time and accuracy of different deep CNN feature extractors on COVID-19 classification.

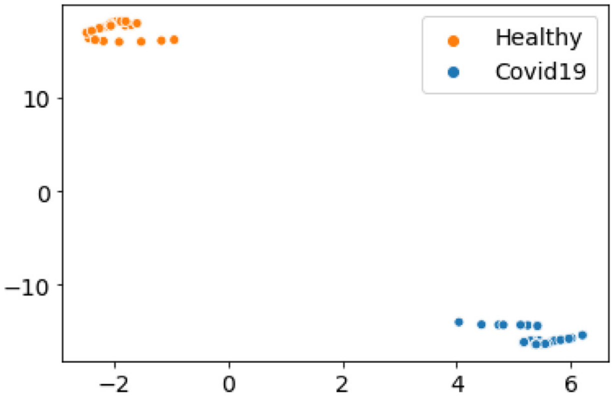

Fig. 10 – The t-SNE visualization results that illustrate the performance of the best model (DenseNet121 and Bagging classifier) on test data.

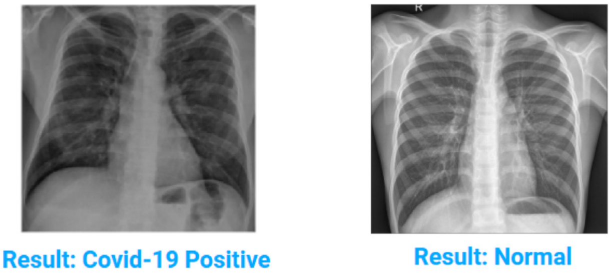

Fig. 11 – Web-based application for automatic detection of COVID-19 pneumonia.

points represent healthy cases and yellow points represent Covid-19 cases). The distribution of the test data from the visualization result in this figure demonstrates the strong ability of the proposed method in classifying each image into the correspond label.

After training a model, the pre-trained weights and models can be used as predictive engine for CAD systems to allow an automatic classification of new data. A web-based application was implemented using standard web development tools and techniques such as Python, JavaScript, HTML, and Flask web framework. Fig. 11 shows the output of our web-based application for COVID-19 pneumonia detection. This web application could help doctors benefit from our proposed method by providing an online tool that only requires uploading an X-ray or CT image. The application then provides the physician with a simple COVID-19 Positive, or COVID-19 Negative observation. It should be noted that this application has yet to be clinically validated, is not yet approved for diagnostic use and would simply serve as a diagnostic aid for the medical imaging specialist.

The proposed method is generic as it does not need handcrafted features and can be easily adapted, requiring minimal pre-processing. The provided dataset is collected across multiple sources with different shape, textures and morphological characteristics. The transfer learning strategy has successfully transferred knowledge from the source to the target domain despite the limited dataset size of the provided dataset. During the proposed approach, we observed that no over-fitting occurs to impact the classification accuracy adversely. However, our study has some limitations. The training data samples are limited. Extending the dataset size by additional data sources can provide a better understanding on the proposed approach. Also, employing pre-trained networks as feature extractors requires to rescale the input images to a certain dimension which may discard valuable information. Although the proposed methodology achieved satisfying performance with an accuracy of 99.00%, the diagnostic performance of the deep learning visual feature extractor and machine learning classifier should be evaluated on real clinical study trials. It is important to note that no statistical inferences were drawn from the obtained results to measure the significance of models' improvements.The final obtained accuracy might be an over optimistic estimate and is not confirmed when the model is tested on independent training and testing sets.

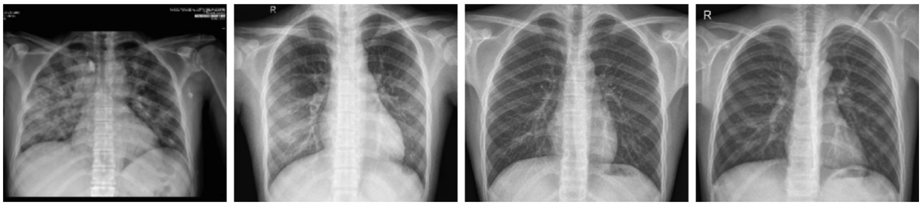

Fig. 9 – Examples of miss-classified cases of COVID-19 dataset.

5. Conclusion

The ongoing pandemic of COVID-19 has been declared a global health emergency due to the relatively high infection rate of the disease. As of the time of this writing, there is no clinically approved therapeutic drug or vaccine available to treat COVID-19. Early detection of COVID-19 is important to interrupt the human-to–human transmission of COVID-19 and patient care. Currently, the isolation and quarantine of the suspicious patients is the most effective way to prevent the spread of COVID-19. Diagnostic modalities such as chest X-ray and CT are playing an important role in monitoring the progression and severity of the disease in COVID-19 positive patients. This paper presents a feature extractor-based deep learning and machine learning classifier approach for computer-aided diagnosis of COVID-19 pneumonia. Several ML algorithms were trained on the features extracted by well-established CNNs architectures to find the best combination of features and learners. Considering the high visual complexity of image data, proper deep feature extraction is considered as a critical step in developing deep CNN models. The experimental results on available chest X-ray and CT dataset demonstrate that the features extracted by DenseNet121 architecture and trained by a Bagging tree classifier generates very accurate prediction of 99.00% in terms of classification accuracy.

In the future, it would be interesting to investigate designing models with weights that are transferred and fine-tuned using a similar domain instead of weights from different domain such as ImageNet dataset.

REFERENCES

[1] Shereen MA, Khan S, Kazmi A, Bashir N, Siddique R. Covid-19 infection: Origin, transmission, and characteristics of human coronaviruses. J Adv Res 2020;24:91–8. https://doi.org/10.1016/j.jare.2020.03.005. URL: http://www.sciencedirect.com/science/article/pii/S2090123220300540.

[2] Lippi G, Plebani M, Henry BM. Thrombocytopenia is associated with severe coronavirus disease 2019 (covid-19) infections: A meta-analysis. Clin Chim Acta 2020;506:145–8. https://doi.org/10.1016/j.cca.2020.03.022. URL: http://www.sciencedirect.com/science/article/pii/S0009898120301248.

[3] Zhang T, Wu Q, Zhang Z. Probable pangolin origin of sars-cov-2 associated with the covid-19 outbreak. Curr Biol 2020;30:1346–1351.e2. https://doi.org/10.1016/j.cub.2020.03.022. URL: http://www.sciencedirect.com/science/article/pii/S0960982220303602.

[4] Guo H, Zhou Y, Liu X, Tan J. The impact of the covid-19 epidemic on the utilization of emergency dental services. J Dental Sci 2020. https://doi.org/10.1016/j.jds.2020.02.002. URL: http://www.sciencedirect.com/science/article/pii/S1991790220300209.

[5] Chavez S, Long B, Koyfman A, Liang SY. Coronavirus disease (covid-19): A primer for emergency physicians. Am J Emerg Med 2020. https://doi.org/10.1016/j.ajem.2020.03.036. URL: http://www.sciencedirect.com/science/article/pii/S0735675720301789.

[6] Rothan HA, Byrareddy SN. The epidemiology and pathogenesis of coronavirus disease (covid-19) outbreak. J Autoimm 2020;109. https://doi.org/10.1016/j.jaut.2020.102433. URL: http://www.sciencedirect.com/science/article/pii/S0896841120300469 102433.

[7] Liu H, Liu F, Li J, Zhang T, Wang D, Lan W. Clinical and ct imaging features of the covid-19 pneumonia: Focus on pregnant women and children. J Infect 2020. https://doi.org/10.1016/j.jinf.2020.03.007. URL: http://www.sciencedirect.com/science/article/pii/S0163445320301183.

[8] WHO, Coronavirus disease (COVID-19) Pandemic, 2020. URL: https://www.who.int/emergencies/diseases/novel-coronavirus-2019.

[9] Shim E, Tariq A, Choi W, Lee Y, Chowell G. Transmission potential and severity of covid-19 in south korea. Int J Infect Diseas 2020;93:339–44. https://doi.org/10.1016/j.ijid.2020.03.031. URL: http://www.sciencedirect.com/science/article/pii/S1201971220301508.

[10] CDC, Coronavirus Infections, 2020. URL: https://phil.cdc.gov/Details.aspx?pid=23313.

[11] CDC, Coronavirus Infections – Transmission electron microscopic image, 2020. URL: https://phil.cdc.gov/Details.aspx?pid=23354.

[12] Wang Y, Dong C, Hu Y, Li C, Ren Q, Zhang X, Shi H, Zhou M. Temporal changes of ct findings in 90 patients with covid-19 pneumonia: a longitudinal study. Radiology 2020;200843.

[13] Kassani SH, Kassani PH, Wesolowski MJ, Schneider KA, Deters R. Depthwise separable convolutional neural network for skin lesion classification. In: 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), IEEE, 2019, p. 1–6.

[14] Hemdan EED, Shouman MA, Karar ME. Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images, arXiv preprint arXiv:2003.11055 (2020)..

[15] Cohen JP, Morrison P, Dao L. Covid-19 image data collection, arXiv:2003.11597 (2020). URL:https://github.com/ieee8023/covid-chestxray-dataset.

[16] Rosebrock Adrian. Detecting COVID-19 in X-ray images with Keras, TensorFlow, and Deep. Learning 2020. URL: https://www.pyimagesearch.com/2020/03/16/detecting-covid-19-in-x-ray-images-with-keras-tensorflow-and-deep-learning/.

[17] Barstugan M, Ozkaya U, Ozturk S. Coronavirus (covid-19) classification using ct images by machine learning methods, arXiv preprint arXiv:2003.09424 (2020).

[18] Cristianini N, Shawe-Taylor J, et al. An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press; 2000.

[19] Wang L, Wong A. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest radiography images, arXiv preprint arXiv:2003.09871 (2020).

[20] Kaggle, Kaggle's Chest X-Ray Images (Pneumonia) dataset, 2020. URL: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia.

[21] Maghdid HS, Asaad AT, Ghafoor KZ, Sadiq AS, Khan, MK. Diagnosing covid-19 pneumonia from x-ray and ct images using deep learning and transfer learning algorithms, arXiv preprint arXiv:2004.00038 (2020).

[22] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, 2012. p. 1097–1105.

[23] Ghoshal B, Tucker A. Estimating uncertainty and interpretability in deep learning for coronavirus (covid-19) detection, arXiv preprint arXiv:2003.10769 (2020).

[24] Hall, LO, Paul, R, Goldgof, DB, Goldgof GM. Finding covid-19 from chest x-rays using deep learning on a small dataset, arXiv preprint arXiv:2004.02060 (2020).

[25] Farooq M, Hafeez A. Covid-resnet: A deep learning framework for screening of covid19 from radiographs, arXiv preprint arXiv:2003.14395 (2020).

[26] Boureau Y-L, Ponce J, LeCun Y. A theoretical analysis of feature pooling in visual recognition. In: Proceedings of the 27th international conference on machine learning (ICML-10). p. 111–8.

[27] Rakhlin A, Shvets A, Iglovikov V, Kalinin AA. Deep convolutional neural networks for breast cancer histology image analysis. In: International Conference Image Analysis and Recognition Springer. p. 737–44.

[28] Guo Y, Liu Y, Oerlemans A, Lao S, Wu S, Lew MS. Deep learning for visual understanding: A review. Neurocomputing 2016;187:27–48.

[29] Quinlan JR. Induction of decision trees. Mach Learn 1986;1:81–106.

[30] Breiman L. Random forests. Mach Learn 2001;45:5–32.

[31] Chen T, Guestrin C. XGBoost. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16, ACM Press, New York, New York, USA, 2016. p. 785–794. URL: http://dl.acm.org/citation.cfm?doid=2939672.2939785. doi: 10.1145/2939672.2939785.

[32] Freund Y, Schapire RE. A desicion-theoretic generalization of on-line learning and an application to boosting. In: European conference on computational learning theory Springer. p. 23–37.

[33] Breiman L. Bagging predictors. Mach Learn 1996;24:123–40.

[34] Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, et al. Lightgbm: A highly efficient gradient boosting decision tree. In: Advances in neural information processing systems, 2017. p. 3146–3154.

[35] Kassani SH, Kassani PH, Wesolowski MJ, Schneider KA, Deters R. Breast cancer diagnosis with transfer learning and global pooling, arXiv preprint arXiv:1909.11839 (2019)..

[36] Khan S, Islam N, Jan Z, Din IU, Rodrigues JJC. A novel deep learning based framework for the detection and classification of breast cancer using transfer learning. Pattern Recogn Lett 2019;125:1–6.

[37] Mehra R et al. Breast cancer histology images classification: Training from scratch or transfer learning? ICT Express 2018;4:247–54.

[38] Kassani SH, Kassani PH, Wesolowski MJ, Schneider KA, Deters R. Automatic polyp segmentation using convolutional neural networks. In: Canadian Conference on Artificial Intelligence, Springer. p. 290–301.

[39] Lu S, Lu Z, Zhang Y-D. Pathological brain detection based on alexnet and transfer learning. J Comput Sci 2019;30:41–7.

[40] Kassani SH, Kassani PH, Wesolowski MJ, Schneider KA, Deters R. Classification of histopathological biopsy images using ensemble of deep learning networks. In: Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering, CASCON '19. USA: IBM Corp; 2019. p. 92–9.

[41] Liu Z, Cao Y, Li Y, Xiao X, Qiu Q, Yang M, Zhao Y, Cui L. Automatic diagnosis of fungal keratitis using data augmentation and image fusion with deep convolutional neural network. Comput Methods Programs Biomed 2020;187 105019.

[42] Liu S, Tian G, Xu Y. A novel scene classification model combining resnet based transfer learning and data augmentation with a filter. Neurocomputing 2019;338:191–206.

[43] Sridar P, Kumar A, Quinton A, Nanan R, Kim J, Krishnakumar R. Decision fusion-based fetal ultrasound image plane classification using convolutional neural networks. Ultrasound Med Biol 2019;45:1259–73.

[44] Zhang W, Zhong J, Yang S, Gao Z, Hu J, Chen Y, Yi Z. Automated identification and grading system of diabetic retinopathy using deep neural networks. Knowl Based Syst 2019;175:12–25.

[45] Dourado Jr CM, da Silva SPP, da Nóbrega RVM, Barros ACdS, Reboucas Filho PP, de Albuquerque VHC. Deep learning iot system for online stroke detection in skull computed tomography images. Comput Networks 2019;152:25–39.

[46] Çinar A, Yildirim M. Detection of tumors on brain mri images using the hybrid convolutional neural network architecture. Med Hypotheses 2020;109684.

[47] Cogan T, Cogan M, Tamil L. Mapgi: Accurate identification of anatomical landmarks and diseased tissue in gastrointestinal tract using deep learning. Comput Biol Meds 2019;111 103351.

[48] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: Efficient convolutional neural networks for mobile vision applications andrew. Rep Pract Oncol Radioth. 2009. doi: 10.1016/S1507-1367(10)60022-3. arXiv:1704.04861.

[49] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional network. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017. 2017. doi:10.1109/CVPR.2017.243. arXiv:1608.06993.

[50] Chollet F. Xception: Deep learning with depthwise separable convolutions. In: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017. 2017. doi:10.1109/CVPR.2017.195. arXiv:1610.02357.

[51] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/CVPR.2016.308. arXiv:1512.00567.

[52] Szegedy C, Ioffe S, Vanhoucke V, Alemi AA. Inception-v4, inception-ResNet and the impact of residual connections on learning. In: 31st AAAI Conference on Artificial Intelligence AAAI. arXiv:1602.07261.

[53] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/CVPR.2016.90. arXiv:1512.03385.

[54] Simonyan K. A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

[55] Zoph B, Vasudevan V, Shlens J, Le QV. Learning Transferable Architectures for Scalable Image Recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/CVPR.2018.00907. arXiv:1707.07012.

[56] Kaggle, RSNA Pneumonia Detection Challenge, 2020. URL: https://www.kaggle.com/c/rsna-pneumonia-detection-challenge.

[57] Ausawalaithong W, Thirach A, Marukatat S, Wilaiprasitporn T. Automatic lung cancer prediction from chest x-ray images using the deep learning approach. In: 2018 11th Biomedical Engineering International Conference (BMEICON), IEEE, 2018. p. 1–5.