Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

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

Detection of COVID-19 in X-ray images by classification of bag of visual words using neural networks

Zahra Nabizadeh-Shahre-Babak [a], Nader Karimi [a], Pejman Khadivi [b,*], Roshanak Roshandel [b], Ali Emami [a], Shadrokh Samavi [a,c]

[a] Department of Electrical and Computer Engineering, Isfahan University of Technology, Isfahan, Iran
[b] Computer Science Department, Seattle University, Seattle, USA
[c] Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada

ARTICLE INFO

Keywords:
COVID-19
Coronavirus
Bag of visual words
Classifier

ABSTRACT

Coronavirus disease 2019 (COVID-19) was classified as a pandemic by the World Health Organization in March 2020. Given that this novel virus most notably affects the human respiratory system, early detection may help prevent severe lung damage, save lives, and help prevent further disease spread. Given the constraints on the healthcare facilities and staff, the role of artificial intelligence for automatic diagnosis is critical. The automatic diagnosis of COVID-19 based on medical images is, however, not straightforward. Due to the novelty of the disease, available X-ray datasets are very limited. Furthermore, there is a significant similarity between COVID-19 X-rays and other lung infections. In this paper, these challenges are addressed by proposing an approach consisting of a bag of visual words and a neural network classifier. The proposed method can classify X-ray chest images into non-COVID-19 and COVID-19 with high performance. Three public datasets are used to evaluate the proposed approach. Our best accuracy on the first, second, and third datasets is 96.1, 99.84, and 98 percent. Since detection of COVID-19 is important, sensitivity is used as a criterion. The proposed method's best sensitivities are 90.32, 99.65, and 91 percent on these datasets, respectively. The experimental results show that extracting features with the bag of visual words results in better classification accuracy than the state-of-the-art techniques.

1. Introduction

In late 2019, a novel coronavirus was discovered in Wuhan, China. The novel coronavirus, named SARS-CoV-2, is highly pathogenic, primarily attacks the respiratory system, and causes a potentially dangerous disease known as COVID-19 [1]. The symptoms of COVID-19 are similar to influenza, and it primarily infects the lungs. However, unlike influenza, COVID-19 spreads faster, and has a higher mortality rate. Due to the worldwide outbreak of COVID-19 in less than three months, in early March 2020, the WHO declared this disease to be a pandemic.

Early diagnosis of COVID-19 helps the treatment staff and prevents further damage to the lungs, and could save the patient's life. There are currently several research initiatives in the field of medical image processing to help support the healthcare staff to diagnose COVID-19 in a shorter time. However, the automatic diagnosis of COVID-19 is a very challenging task due to multiple reasons. First, there are similarities between COVID-19 and other respiratory diseases in terms of their impact on the lungs. Furthermore, the number of existing COVID-19 images is relatively small, directly impacting the successful training of complex machine learning models. Moreover, given the low-resolution of X-ray images in the datasets, the extraction of distinctive features is particularly challenging.

In this paper, to address the three mentioned challenges, a two-step process is proposed: first, a bag of visual words is used to extract relevant features. Then the neural networks are used to classify images into two groups of non-COVID-19 and COVID-19.

The rest of the paper is organized as follows. In Section II, the literature is reviewed. In section III, the proposed method is explained. Experimental results are presented in Section IV. Section V is dedicated to some concluding remarks.

[*] Corresponding author.
E-mail address: khadivip@seattleu.edu (P. Khadivi).

https://doi.org/10.1016/j.bspc.2021.102750
Received 3 September 2020; Received in revised form 1 March 2021; Accepted 9 May 2021
Available online 14 May 2021

2. Literature review

Several approaches are proposed to facilitate the diagnosis of COVID-19 based on medical images. These techniques can be categorized as classification and segmentation. In the following subsections, these techniques are reviewed.

2.1. Classification-based approaches

Accurate classification of patients' lungs X-ray images can support early quarantine of patients and limit its rapid spread. The two primary goals for COVID-19 classification include detecting positive COVID-19 cases [2–4], and identifying healthy lung images from diseased ones such as those with viral, bacterial, and COVID-19 infections [5–8]. In these approaches, two types of chest X-ray, or Computed Tomography (CT) images, are used.

In [2], Sethy and Behera propose a classification approach that consists of two steps: feature extraction using Resnet50 and classification using Support Vector Machine (SVM). Different Convolutional Neural Networks (CNNs), such as VGG16, AlexNet, GoogleNet, and Resnet, are explored for feature extraction, and the best accuracy is achieved when the features of Resnet50 are used for classification [2]. The main shortcomings of CNNs are that they are prone to losing the spatial information between images. Furthermore, since CNNs have a large number of parameters, they need large datasets for training purposes. To address these shortcomings, Afshar et al. proposed COVID_-CAPS based on capsule networks [3]. This network has fewer parameters than other CNNs, and by using capsule layers, the information between features is also extracted, producing better accuracy as compared to CNNs [3]. In [4], three-dimensional CT-scan (3D-CT) is used to detect COVID-19, and high accuracies are reported. In [4] at first, a pre-trained UNet is used to segment the lungs region from the chest images, and then these *regions of interest* (ROI) are fed to a 3D deep neural network for classification. With this approach, they obtain good performance in detecting COVID-19 without annotating the lesions in the images. Since extracting features is a crucial stage for classification, different feature-extractors are tested in [9], such as Grey Level Co-occurrence Matrix (GLCM), Local Directional Pattern, Grey Level Run Length Matrix, Grey-Level Size Zone Matrix (GLSZM), and Discrete Wavelet Transform. For classifying these features, SVM is trained. The best result was obtained when GLSZM features were used.

In some cases, the accurate diagnosis of COVID-19 from other diseases is vital. There are different types of images in datasets that show infected lungs, but they are not COVID-19. In [6], Wang and Wong proposed a new architecture, called COVID_Net, which consists of projection-expansion-projection-extension blocks. With this architecture, chest X-ray images in the dataset can be classified into three healthy, non-COVID-19, and COVID-19 groups. For controlling the spread of COVID-19, screening a large number of cases is required. Due to this, a new large CT dataset is collected by the authors of [7]. For classifying these images, first, the ROI part of the lungs in each image is detected. Then, in the second step, transfer-learning is used in an Inception network to extract features. Finally, all images are classified with a fully connected network (FCN) [7]. Ying et al. [8] also collected a dataset of 3D-CT images, including healthy, bacterial, and COVID-19 images. They first extract the ROI and feed them into a deep network (DeepPneumonia). This network predicts the class of each image slice. In the last step, the results of slice classification are aggregated, and the class of image is predicted.

In [10], Sethy et al. extracted features with two methods: learning-based, and image processing-based. They used different CNN models such as Resnet50 to extract features and SVM as their classifier for learning-based methods. For image processing methods, algorithms such as local binary patterns (LBP), histogram of oriented gradients (HOG), and Grey Level Co-occurrence Matrix (GLCM) are selected to extract features, and SVM is used for the classifier. As mentioned in [10], the SVM is selected as a classifier instead of deep learning models because the latter needs more data for training and validation. Their experimental results show that the deep features result in better performance. Narin et al. in [11] use five pre-trained models such as ResNet50, ResNet101, ResNet152, InceptionV3, and Inception-ResNetV2 for separating four different categories of X-ray images. For this purpose, three binary classifications are implemented. Their experimental results show that the ResNet50 model results in better accuracy.

In some cases, due to the small dataset of COVID-19 images, the deep neural networks are not proper. In some articles, identified models are changed to simple models. In [12], the Inception Net V3 is simplified and named truncated Inception net. By using this model on six datasets, the average accuracy equal to 98.77 is achieved. In [13], for detecting COVID-19, Inception V3 is used with the transfer learning technique. Transfer learning is a technique that is used in situations that the dataset is small. This technique is more applicable in COVID-19 detection. For example, in [14], this technique is used in the Xception model, and a good performance is achieved.

In some cases, training from scratch results in better performance. In [15], different modes of feature extraction, training from scratch, feature extraction via transfer learning, and hybrid feature extraction via fine-tuning, are explored. Experimental results show that training from scratch is the best approach. Sahinbas and Catak in [16] consider different pre-trained CNN models such as VGG16, VGG19, ResNet, DenseNet, and InceptionV3. By using these models, the best accuracy is achieved when the VGG16 is used. In [17], for classifying the X-ray images, a multi-step method is proposed. At first, a preprocess block is applied for removing Gaussian noise. After that, the lung region is segmented from the images. In the third block, the CNN model extracts the features, and FCN is used to classify the images. In [18], the authors prepare a 5 K dataset from the public dataset and evaluate different CNN models such as ResNet18, ResNet50, SqueezeNet, and DenseNet-121. They examine different thresholds, and their best result is achieved when the model is SqueezeNet, and the threshold is 0.15. In [19], the authors conclude that using pre-trained models such as AlexNet, GoogLeNet, SqueezeNet compare to the new model result in better performance.

2.2. Segmentation-based approaches

Segmentation of the infected regions of the CT images can help the classifier to achieve higher accuracies. In [20], Yan et al. use a feature variation block to enhance the edge of infection areas adaptively using a new deep CNN. In their architecture, features of different scales are fused to identify the infected parts with different shapes and sizes. Fan et al. [21] proposed the lung Infection Segmentation Deep Network (Inf-Net), using a parallel partial decoder. In their work, features of different levels are fused to generate the global map.

Multitask deep learning models are also used in the literature for image classification and segmentation purposes. Amyar et al. [22] proposed a multitask model that jointly separates COVID-19 from non-COVID-19 images and segments the lesion regions in CT images. Their network architecture consists of an encoder and two decoders for reconstruction and segmentation, and a multi-layer perceptron for classification. The purpose of their model is to consider useful multitask information to perform both segmentation and classification.

Extracting features from the dataset is one of the essential steps for classification tasks. As mentioned above, in most of the recent articles, features are extracted by neural networks. Since in new problems the datasets are small, the deep neural networks face with the problem of overfitting. There are different approaches to address this problem. In some cases, extracting handcrafted features can help the neural networks to classify the data more accurately. Therefore, in this paper, features are extracted by the bag of visual words for classification.

3. Proposed method

This paper proposes a three-stage approach that takes a chest X-ray image as the input and determines the class probability as the output. The proposed approach includes (1) preprocessing, (2) creating a Bag of Visual Words (BVW), and (3) classification. The three-stage block diagram of the approach is illustrated in Fig. 1. These stages consist of subblocks, which will be explained in the following subsections.

3.1. Preprocessing

The lung images of COVID-19 patients contain discriminative information, which, if extracted, can be used to classify COVID-19 from non-COVID-19 cases. The information is embedded in the intensity distribution of the images and needs preprocessing to be revealed. One of the significant issues with the COVID-19 chest X-ray datasets is the wide variation of images due to the patients' age, configuration of the X-ray capturing devices, and positioning of patients in the field of view of the X-ray devices. Hence, the foreground and background areas in the X-ray images, and the pixel intensity distribution of images have wide varieties. Furthermore, X-ray images are typically low resolution. One of the sample images is shown in Fig. 2c. Therefore, an appropriate preprocessing method is required to help the feature extractor to focus on the image's significant areas and extract distinctive features. For this purpose, in the preprocessing phase, histogram matching and intensity improvement methods are used. These methods are described below.

3.1.1. Histogram matching

The range of pixel intensities and the size of the images' background area are different in images captured by different devices. Therefore, we opt to use histogram matching as an effective method to decrease the influence of different recording devices [23]. The average histogram of the training images is used to bring all the dataset images into a unified range of intensities. In this method, the average histogram of all images in the training dataset is calculated using the following equation:

$$p_r(r_j) = \frac{1}{N}\sum_i \frac{n_{ij}}{n_i} \tag{1}$$

where r_j, n_{ij}, n_i, and N represent the j^{th} level of intensity, number of pixels with intensity r_j, the number of all pixels in the i^{th} image, and the number of all images in the training set, respectively. This histogram represents the average distribution of all images in the training set. The first step in performing histogram matching is to calculate the average histogram's cumulative distribution function (CDF). The CDF of each image in the training and test sets is calculated using the following equation:

$$P_r(r_k) = \sum_{j=1}^{k} p_r(r_j) \tag{2}$$

For all intensity levels in an image, the distance between the CDF value of one intensity level in the image and the average CDF value is calculated. In the end, the minimum value is selected for the intensity level. Fig. 2 shows an example of applying histogram matching. The mean histogram is shown in Fig. 2a. The mean CDF and CDF of an image before and after the transformation are illustrated in Fig. 2b. The original image, Fig. 2c, and the transformed image, Fig. 2d, are also shown. There are many differences in intensity levels between images in the dataset. The red curve spike means that for matching the histogram of the image to the mean histogram of train images, there are no intensity levels before 65 in the output image (image after transfer).

3.1.2. Intensity improvement

Many of the images in the dataset have low contrasts. An example of a low contrast chest X-ray is illustrated in Fig. 3a. A transformation that improves image contrast and sharpens its edges is thus useful as part of preprocessing. The image contrast is improved by using contrast stretching techniques that expand the image histogram to the desired range. There are two types of stretching: linear and nonlinear. Since it is required to emphasize high gradient points in our work, the nonlinear stretching model is applied. In the linear method, all intensities are changed. In the nonlinear method, only parts of the intensity range are modified. In our selected method, one percent of each image's pixels from the low-intensity range of images are transferred to 0 intensity level, and one percent of pixels from the high-intensity range of images are saturated to 255. By applying the nonlinear method, the intensity range of the image covers the whole intensity range [0255], and the contrast of edges in the image will be increased [24]. In Fig. 3, an original image, the output of histogram matching, and the output of the intensity improvement are shown. The corresponding histogram of each image at each step is also included in Fig. 3. The histogram of the image in the last step compared to the histogram of the previous step is similar in shape, and only the intensity range is extended. It will be shown that these preprocessing steps will help the classification process.

3.2. Creating bag of visual words

Bag of visual words (BVW) is a technique that can be used for image classification. In BVW, an image is represented by the frequency of its features [25]. For implementing BVW, two steps, extracting features and creating a dictionary, are considered. This approach allows us to perform image classification based on both individual image features and patterns associated with groups of features. The X-ray images of the patients with COVID-19 infection show that the infected regions have higher gradients as compared to their neighboring areas, but inside the regions, there is less texture. A sample non-COVID-19 image and a COVID-19 case are shown in Fig. 4.

In the BVW phase, *keypoints* and *descriptors should be considered.* Keypoints are those points in an image that standout despite basic image transformations such as shrinking or expansion. Descriptors are descriptions of these keypoints. In the context of this work, it is essential to select a keypoint extraction method that covers the entire lung region. Furthermore, selecting a descriptor extractor that focuses on significant gradient points and local features is helpful.

Several techniques such as Histogram of Oriented Gradients (HOG) [26], Speeded Up Robust Features (SURF) [27], Harris Corner Detector [28], and Scale-Invariant Feature Transform (SIFT) [29] have been used to extract local descriptors. Our goal is to apply an algorithm to extract keypoints and descriptors. Then a dictionary that contains features from normal and abnormal lung regions is created.

Given that the lung's infected areas tend to be less textured and look whiter, our goal is to guide the system to select keypoints in texture-less regions to improve detection results. To implement keypoints selection, two primary methods of grid-based and detector-based do exist. In the grid-based approach, the points are selected according to a user-defined

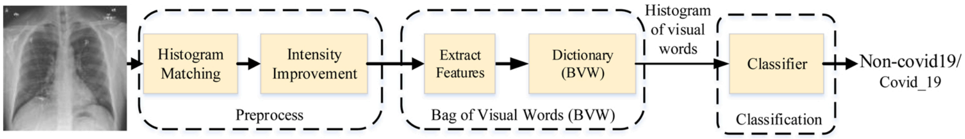


Fig. 1. Block diagram of the test phase of the proposed method.

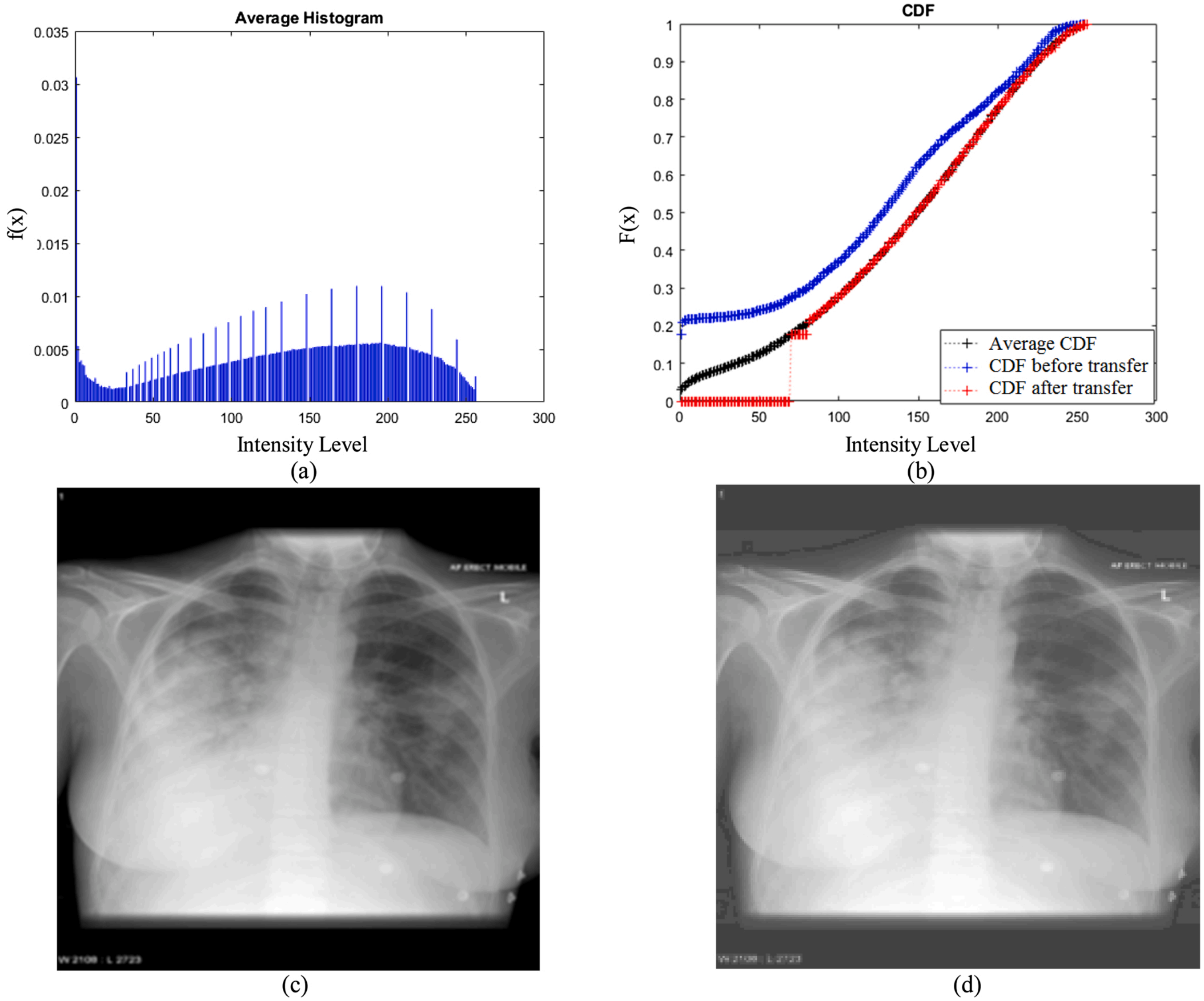


Fig. 2. (a) The average histogram of train images, (b) CDF of image before transformation (blue), CDF of the average histogram (black) and CDF of the image after transformation (red), (c) original image with black background, (d) the output image after transformation.

grid step, starting from the top left corner of the image, and traversing to the bottom right. In the detector-based approaches, descriptors are extracted from the image, and keypoints will be a subset of the extracted descriptors.

In this paper, SURF is used for extracting descriptors because, similar to SIFT, it extracts local and distinctive properties such as gradient information in vertical and horizontal directions around each keypoints. At the same time, the SURF algorithm is faster and more robust against different image transformations. SURF relies on the Hessian matrix's determinant for selecting the location and the scale [27]. A descriptor vector is instantiated for a predefined size window, which is placed around each keypoint. The process of keypoint selection and descriptor vector definition is performed for all images in the training dataset. The descriptor vector contains the gradient information for each image patch and represents a visual word in the dictionary. These descriptor vectors are extracted from different image scales in SURF. These scales are calculated with the patch sizes, which the user defines. The number and values of scales are essential factors in extracting distinctive descriptors. In Section 4 (experimental results section), the effect of methods for selecting keypoints and different patch sizes will be discussed.

In the second step, when all the patches are identified and related descriptors are collected, the dictionary will be created. However, using all the descriptors for creating a dictionary is inefficient. Finding the closest visual words among too many descriptors results in an expensive search process. Since there is no ground truth for visual words in the dictionary, an unsupervised learning method, such as clustering, is applied to descriptors to select the strongest descriptors. Each descriptor has a score that is defined by the extractor. Then descriptors are clustered to form visual words in the dictionary.

In this paper, the K-means clustering algorithm [30] with random cluster centers and iterative feature categorization is chosen. After each iteration, the center of clusters is updated. Iterations are repeated until the sum of distances of all descriptors from their cluster centers becomes less than a threshold. Finally, when all the descriptors are categorized, each cluster's center will be considered a dictionary word, and the dictionary is formed.

3.3. Classification

The dictionary will be used to generate a histogram of descriptors for each training and test image of the dataset. This histogram is used as an input to the classifier. The dictionary's size, and subsequently, the histogram vector's length is determined by the number of clusters in the previous step. It is observed that the number of clusters dramatically

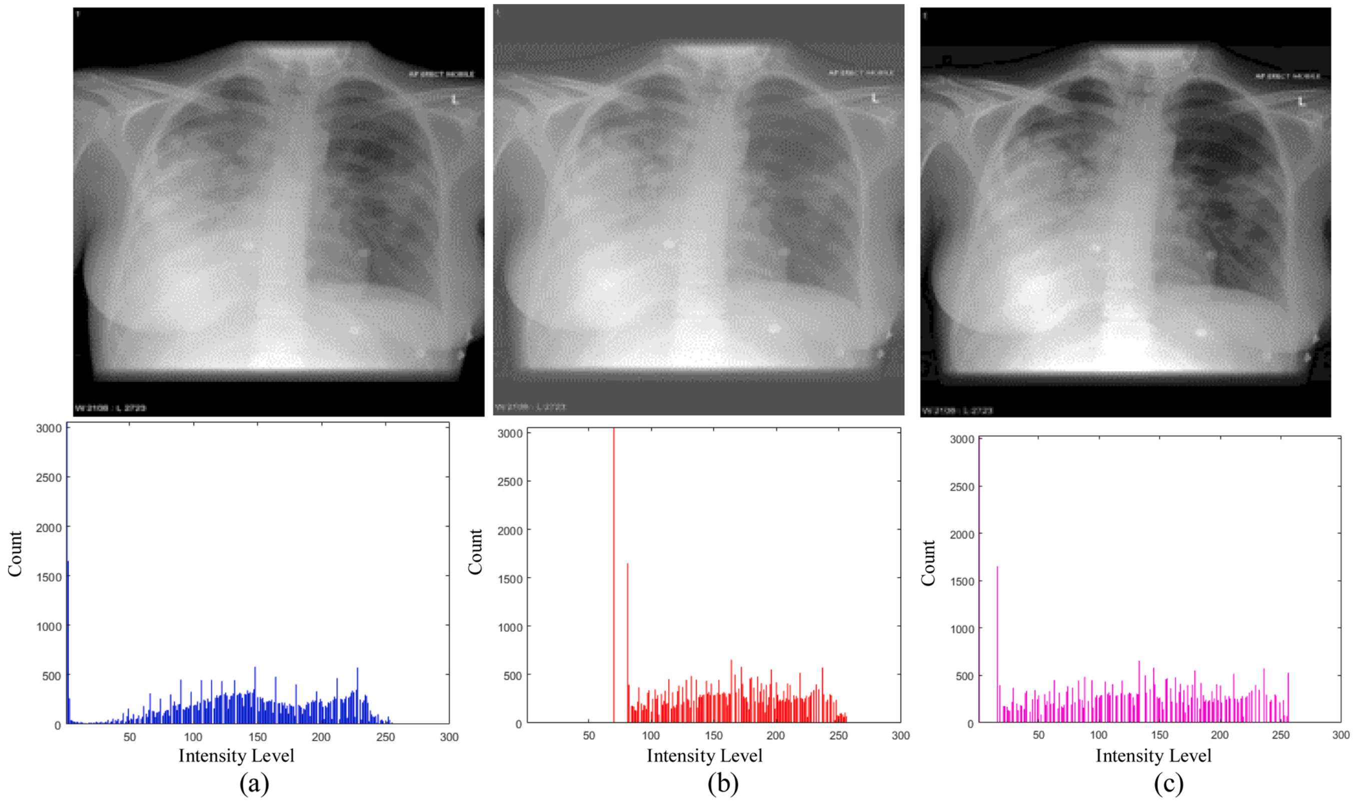


Fig. 3. (a) The original image and it histogram, (b) the image after histogram matching block and its histogram, (c) the image and its histogram after intensity enhancement.

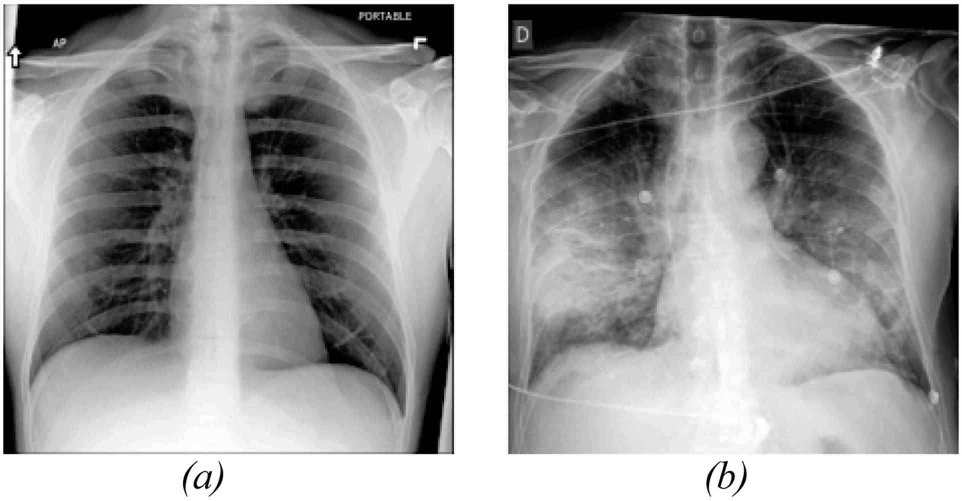


Fig. 4. (a) the image of non−COVID19, (b) the image of a COVID-19 case.

affects the accuracy of the classifier. Therefore, the overall system performance is directly related to the number of clusters. The effect of the length of the dictionary will be discussed in the Swection 4 (experimental results section). For our proposed method's classification stage, any adaptive machine learning classifier or deep learning method can be used. Different classifiers are tested, and the best results are achieved when a Fully Connected Network (FCN) is used.

4. Experimental results

Public datasets used in [2,3] and [18,19] are selected to evaluate our proposed methods. In the following, the details of implementation and datasets and the effects of each parameter are explained.

4.1. Implementation details

In this paper, for implementing the proposed approach, different classifiers are implemented in Python programming langiage. The configuration of the system which is used in our experiments is as follows:

- CPU: Intel(R) Core(TM) i7−7700 K CPU @ 4.20 GHz
- GPU: NVIDIA GeForce GTX 1080 Ti

4.2. *Dataset*

The proposed method is evaluated on three datasets [31–33]. Dataset 1 consists of three datasets [31]. Due to the prolongation of the COVID-19 disease, more X-ray images become available, and hence, different versions of the dataset are created. In this paper, version 3 of the dataset is used. In this version, there are three groups of images: healthy lung X-rays, diseased but non-COVID-19 lung images, and lung images of COVID-19 patients. The dataset contains 152 samples of COVID-19 and over 13,000 samples of the other two classes of X-rays. For extracting a balanced number of features for the training set, 152 samples from each of the three categories of healthy, non-COVID-19, and COVID-19 are selected. Then, for binary classification, healthy and non-COVID-19 cases are merged and considered as a non-COVID-19 class. With these 456 samples, our proposed approach can classify the test images with high accuracy. The second dataset [32] includes 2040 images of the non-COVID-19 category and 1143 images of the COVID-19 category. In collaboration with a group of physicians, a team of researchers from different countries has created this database of the chest X-ray images for COVID-19 positive cases along with normal and viral pneumonia images. The third dataset [33] is named the 5 K dataset. This dataset includes 184 COVID-19 images and 5000 non-COVID-19 images. The information about these three datasets is reported in Table 1. For tuning the hyperparameters, the dataset [31], which is selected first, is used.

4.3. *Effect of different stages of the proposed model*

In this paper, the classifier has two roles. In the first role, the classifier's accuracy is used to select the preprocessing methods and the best value for the bag of visual words parameters. For this purpose, the classifier is fixed to the SVM model, and only the values of parameters are changed. In the second one, the classifier's accuracy is used for selecting the best classifier for our problem. In the following, the effect of each phase of our proposed method will be explained.

4.3.1. *Preprocessing stage*

Some research works use ROI detection to remove misleading information and improve the accuracy of classification. In this paper, instead of extracting this information with complicated methods, the vital information is highlighted by applying simple preprocessing methods. It is shown that utilizing suitable preprocessing operations compensates for the limitations and challenges that may exist in the dataset and can lead to comparable performance as with complex state-of-the-art approaches. The effects of the two procedures, which are mentioned in the preprocessing phase of Fig. 1, are explained in the following subsections.

4.3.1.1. Histogram matching procedure. Experimental results show that images recorded with different devices result in different intensity distributions, affecting classification accuracy. By performing histogram matching, the intensity distribution of all images will be device invariant. We used histogram matching for each image in the dataset by transferring its histogram to the average histogram of all images in the training dataset. Then the final classification results on our test dataset are compared with the two sets (with and without histogram matching). The confusion matrices of these two tests are reported in Table 2. These results show that histogram matching improved the sensitivity from 0.838 to 0.903, but specificity decreased by 0.15, from 0.98 to 0.965. Given the unknown and complex nature of the COVID-19, we argue that detection is essential. The proposed histogram matching results in higher sensitivity and improves the chances of detecting diseased lungs.

Table 1
The details of dataset.

Dataset	COVID /Non-COVID Train	COVID /Non-COVID Test	Version
[31]	152/13,482	31/200	3
[32]	572/1020	571/1021	January-2021
[33]	84/2000	100/3000	January-2021

4.3.1.2. Intensity improvement procedure. As shown in Fig. 3, image contrasts, especially along the edges, are improved by mapping the intensities to a broader range. Our experimental results confirm that this process affects the feature extractor to generate distinctive features, which in turn results in better accuracy of the classifier. Two tests, with and without contrast enhancement, are performed to investigate the effect of this step. Since the descriptors include the gradient information, edge sharpening results in extracting distinctive descriptors. The accuracies of these tests are 0.95 and 0.94, respectively. Hence, the contrast enhancement increases the accuracy by one percent.

4.3.2. *Bag of visual words stage*

The output of the bag of visual words phase is a dictionary, which is the core of the proposed approach. The extracted features directly affect classification accuracy. For this step, several parameters should be considered for tuning the approach. They include:

- Keypoint selection process (Detector method or grid step)
- Patch size
- Strongest features
- Dictionary size

As mentioned earlier, detector-based and grid-based approaches can be used to select the location of descriptors in an image. The histogram associated with these extracted descriptors is used for the classification step. Since many of the images in the dataset are of low resolution and the infected areas are not highly textured, it is argued that selecting distinctive regions based on the detector approach is inefficient. Specifically, the detector approach may miss some of the critical keypoints. This problem is resolved by selecting the grid-based method for selecting descriptors. In the grid-based approach, the points are selected according to a specified grid step. The entire lung region is then sampled, and the descriptors are extracted. As shown in Fig. 5, the number of keypoints selected with the detector method (Fig. 5b) is less than the number of selected keypoints with a grid-based approach (Fig. 5a). It is also observed that the keypoints selected with the detector method will not cover all of the lung's regions where COVID-19 may be present. Therefore, using a detector-based method, important information may be lost. Extracting descriptors using detector-based methods results in visual words in a small part of the intensity spectrum. In contrast, the grid-based method extracts more visual words, where some of them may not be useful. The extracting descriptor's function returns two values: descriptors and their scores. The score can be used as a threshold to prune irrelevant or less-important descriptors from the clustering step.

As mentioned before, SURF is used as a descriptor extractor. One of the advantages of algorithms like SURF for this problem is that these algorithms extract descriptors from different scales and provide a

Table 2
Confusion matrices for tests with and without histogram matching.

With histogram matching	Predict value		
		non-COVID	COVID
Actual value	non-COVID	0.965	0.035
	COVID	0.097	0.903
Without histogram matching	Predict value		
		non-COVID	COVID
Actual value	non- COVID	0.980	0.020
	COVID	0.161	0.839

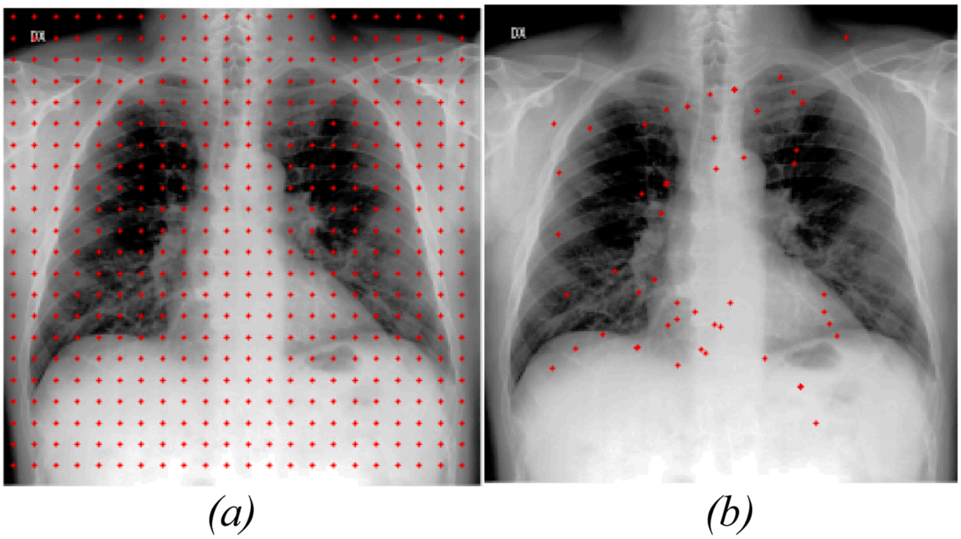


Fig. 5. Keypoints selected by (a) grid-based method, and (b) detector-based method.

different receptive field. Since infected regions in the lungs are of various sizes, selecting SURF for extracting descriptors is appropriate. Choosing proper scales is essential in this work. For this purpose, three patch sizes of 64, 96, and 128 are used. These patch sizes result in the best accuracy for the image sizes of our dataset. With these patches, descriptors are calculated around selected keypoints in three scales. The experimental results show that small size for patches results in extracting features that are not distinctive.

As mentioned before, in SURF, in addition to the descriptors, a score is calculated for each descriptor, which determines the importance of the descriptor. In this process, the parameter which uses this information is the strongest feature. With this parameter, a threshold can be determined to remove weaker descriptors. By setting this parameter, the percentage of descriptors with high scores is selected. The value of this parameter can be adjusted in the range of [0,1]. Experimentally, in our work, this parameter is set to 0.8. Larger values result in outlier descriptors to be added to the dictionary. In comparison, smaller values result in fewer descriptors to be used as input to the clustering step. The reported results in Table 3 show the effect of this parameter. It can be seen that too small and too large values for this parameter harm the accuracy, sensitivity, and specificity of classification.

Using all of the extracted descriptors is not always efficient since the processing cost for finding the closest descriptors is relatively high. Therefore, in this paper, for reducing the number of descriptors and clustering them, K-means clustering is used. Determining K (the number of clusters) is a common challenge with this approach, and the specific value will change based on the specific images in the dataset. The number of clusters determines the size of the dictionary and the length of the histogram that represents each image. Using a small value for the number of clusters results in clusters that contain samples with different distributions. This subsequently results in high distance errors and small-size dictionaries. Our experimental results also show that the extracted histograms are not distinctive enough, and the class accuracy is low when using a small dictionary. On the other hand, using a large value for the number of clusters results in the clustering of similar descriptors in different clusters, which subsequently results in higher processing costs. Experimental results also show that the extracted histogram from an extensive dictionary does not necessarily result in high accuracy. Our algorithm is tested with different dictionaries, varying the size between 250 and 6000.

Table 3
Accuracy, sensitivity and specificity of classification for different values of strongest features.

Strongest Features	Accuracy	Sensitivity	Specificity
0.5	0.9134	0.7096	0.945
0.8	0.9523	0.8709	0.965
1	0.9220	0.8387	0.935

The confusion matrices of experimental results are shown in Table 4. The results in Table 4 show the SVM classifier's output and indicate that dictionaries that are too small or too large cause low accuracies. Dictionaries that are too small result in insufficient visual words for extracting distinctive features. Dictionaries that are too large create useless information. Experimentally it is found that the best specificity belongs to a dictionary of the size 6000. The best sensitivity is achieved when the dictionary size is 5000. Given that the algorithm's sensitivity criterion (number of correctly detected COVID-19 cases) is of primary importance, the dictionary's size is selected to be 5000.

To create a better visualization of extracted histograms and their distinctiveness, the most useful visual words in the histograms of two images, one of non-COVID-19 (blue curve) and one of COVID-19 (red curve) class, are shown in Fig. 6. As seen, the blue bins are distinctively different from the red bins: hence the classifier can accurately distinguish COVID-19 images from the non-COVID-19 cases.

4.3.3. Classification stage

Once a histogram vector containing all descriptors is formed in the previous step, a classifier can be used to make class predictions. Any machine learning model, such as SVMs, neural networks, and KNN, can be used as the classifier. Finding the right classifier is critical for the accuracy of our results. Consequently, several classifiers are experimented with to identify the one with the best results. For each classifier, different settings can be used for the model parameters. In this section, each classifier and the specific parameters that were adjusted for analysis will be described.

From the implementation perspective, in Python Scikit-learn package, for Logistic Regression, the most important parameters related to our work are "penalty" and "solver." Penalty defines the type of regularization, while solver defines the optimization function. Since there is no overfitting problem in this classifier, the penalty is set to 'None.' The solver parameter defines the optimization function. It is also found that the 'sag' optimization produces the best accuracy and fast convergence. In this paper, we also tested Linear SVM, which has a linear kernel. For the K-Nearest Neighbors model, the number of neighbors is varied. For this application, KNN produces its best accuracy for ten nearest neighbors. Linear Discriminant Analysis (LDA) is a technique for dimension

Table 4
Confusion matrix for different dictionary sizes.

Dictionary Size 250	Predicted value			Dictionary Size 5000	Predicted value		
		non-COVID	COVID			non-COVID	COVID
Actual value	non-COVID	0.92	0.09	Actual value	non-COVID	0.96	0.04
	COVID	0.23	0.77		COVID	0.129	0.871
Dictionary Size 2000	Predicted value			Dictionary Size 6000	Predicted value		
		non-COVID	COVID			non-COVID	COVID
Actual value	non- COVID	0.92	0.08	Actual value	non-COVID	0.97	0.03
	COVID	0.13	0.87		COVID	0.19	0.81

reduction. It is also a classifier that automatically reduces the dimension and fits class conditional densities to the data with a linear decision boundary that uses Bayes' rule. Singular Value Decomposition (SVD) as a solver is recommended for data types with many features in this classifier. Gaussian Naïve Bayes algorithm can also be used as a classifier. In this algorithm, the parameters are updated online with a partial fit. For updating variance, the "var_smoothing" parameter was set to 2×10^{-5}. Decision Trees can be used for classification and regression. In this paper, a classifier is needed. For this classifier, the parameter which specifies the number of features to be split is set to 'auto.' Random Forest is a machine learning algorithm. Each tree in a forest is fit to sub-samples of the dataset. The number of trees in the forest is set to 100. All these classifiers can be invoked from the Svikit-learn library in Python. Table 5 presents the results for different classifiers that are used for experimentation. As shown, the neural network results are superior to the other classifiers. More specifically, a fully connected network (FCN) is used. The reason for using FCN is that during the training process, the FCN learns to emphasize the more essential features in the bag of visual words and how to combine them to get the best result. Besides this advantage, having adjustable hyperparameters also makes this a superior approach.

Selecting the appropriate architecture is one of the first steps in designing and implementing a neural network model. The number of layers, number of neurons in each layer, loss function, and the learning rate are important hyperparameters, which affects the accuracy of the classifier. Despite their promising performance, there is still no solid analytic approach to determine these settings, for neural networks [34], and we should rely on experimental results. Different architectures have experimented with intending to keep the network small enough to avoid overfitting due to the low number of samples. Ultimately, the selected architecture consists of three fully connected layers. The first two layers have 100 neurons, and the last layer, which announces the probability of COVID-19 class, has one neuron. Given that the goal is to classify the data under two classes and the imbalanced nature of samples, the weighted binary cross-entropy was used for the loss function. Although the weights for the two classes are set, the predicted values for the non-COVID-19 class have lower errors, but the error of predicted values for the COVID-19 class is high. A combination of binary cross-entropy and F1 score is used for the loss function to address this problem. The model loss is calculated according to the following equation:

$$Loss = \frac{BCE + (1 - F_1 score)}{2} \tag{3}$$

where BCE represents the Binary Cross Entropy value. Note that since the weights of the two parts of the loss function (i.e., BCE and F1 score) are equal, the total value is divided by 2. The F1 score is calculated based on the following equation:

$$F_1 score = \frac{2TP}{2TP + FN + FP} \tag{4}$$

Where TP, FP, and FN are the ratios of the true-positive, false-positive, and false-negative cases, respectively. As shown in Eq. 4, the true negative, which determines the probability of non-COVID-19 correct predicted value, does not affect the F1 score value, and only TP is used. Hence, to decrease loss value, the model tries to learn COVID-19 samples better.

The values of extracted features from images of the two classes are very close to each other. Therefore, to find the best curve for separating these two classes, the step of changes should be very small. Two parameters, namely the learning rate and its decay value, which determine the changing step, are essential. For three datasets, different values for these two parameters are selected.

4.4. Execution time

For calculating the execution time, the duration for creating a bag of visual words and classifier should be calculated. For this purpose, the average running time for creating a bag of visual words is 0.06 s. The average running time for FCN classification is equal to 78 μs. The Python code runs on GPU, and its execution time compared to the BVS, which runs only on the CPU, is very low.

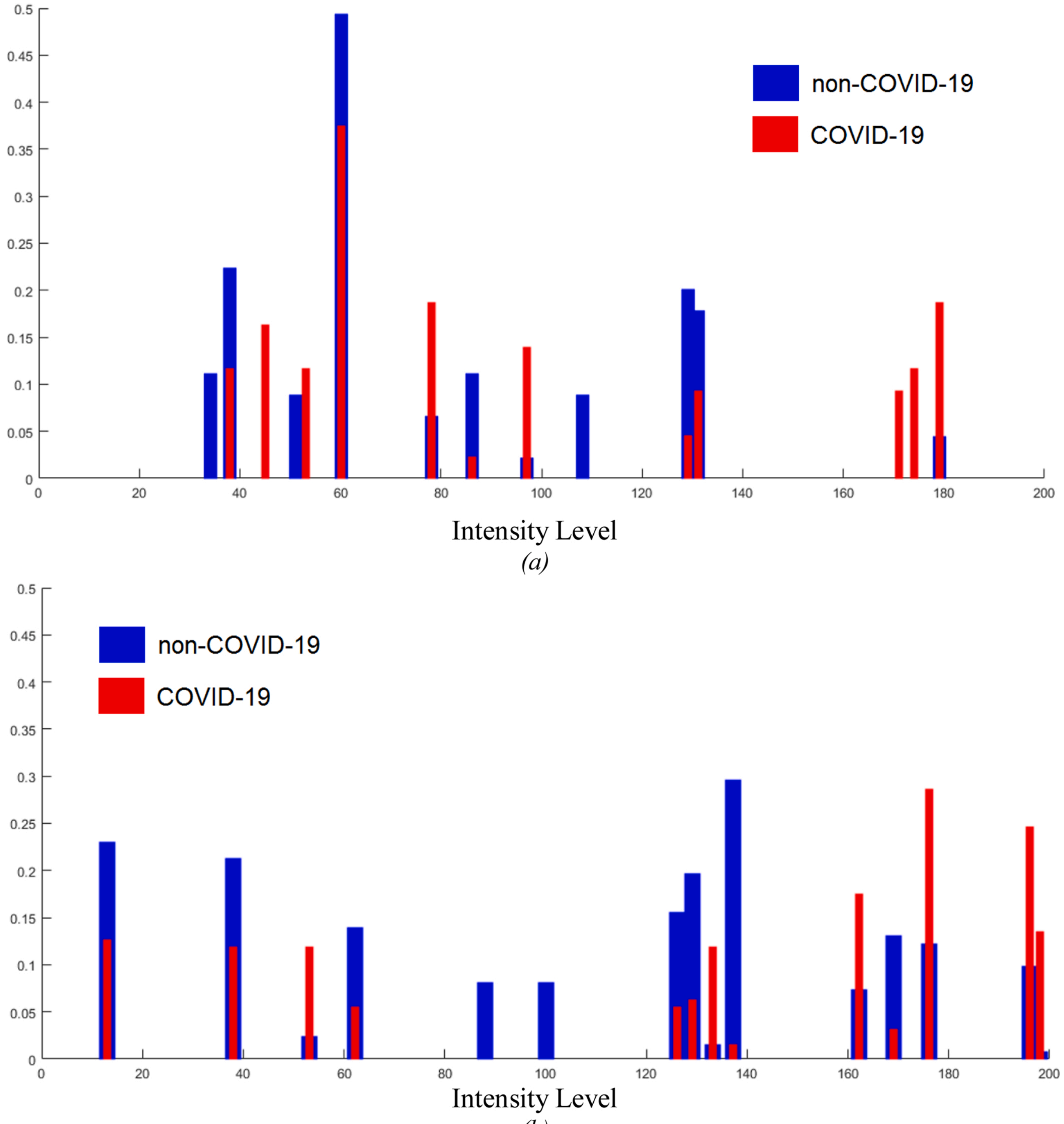
5. Comparison with other methods

In this section, our experimental results are compared with state of the art solutions. For a better overview, the details of different methods used for COVID-19 detection are reported in Table 6.

Since COVID-19 disease spread worldwide and became a pandemic, different datasets from different countries are published publicly, and the number of their samples has changed over time. Due to this issue, researchers have used datasets with different numbers of COVID and Non−COVID samples. In Table 6, the information about the number of COVID and Non−COVID samples in the test set, the applied feature extraction method, the classifier type, and the method's accuracy are reported. By using this table, our method can be compared with the recently published papers in this field.

As mentioned before, the proposed method is evaluated on three datasets. In the following paragraphs, the results of the proposed method on these datasets will be explained.

The first dataset is used in [2,3]. Afshar et al. [3] use Capsule Network for feature extraction and classification. We have used the dataset used by Wang et al. [6] for our work. The size of the dataset containing COVID-19 X-ray images has increased over the past few months. Hence, the dataset now has multiple versions. The authors of [3] do not specify the specific version of the dataset used for analysis. Sethy and Behera [2] use two datasets and rely on deep networks for feature extraction and SVM for classification. Our results are compared with the results presented in [2] and [3] on the first dataset. Comparisons are shown in Table 7. By using 0.5 as a threshold for converting the outputted probability to a class number, the proposed method demonstrates higher accuracy and specificity compared to [3]. In this experiment, different thresholds are tested. When the threshold is set as 0.4, the achieved accuracy, sensitivity, and specificity are 95.7 %, 90.32 %, and 96.5 %, respectively. In this situation, the sensitivity, which represents the accuracy of COVID-19 detection, is improved. Using either of the two thresholds, our approach results in better accuracy and specificity but lower sensitivity than [2].



(a)

(b)

Fig. 6. Histograms of the most frequent visual words for (a) two overlapped histograms of inside of the lungs, and (b) outside regions of lungs. Visual words from COVID-19 images shown by red, and non-COVID-19 by blue.

For generalization, the proposed method is also evaluated on two other datasets. Since there are no train and test splits in the second dataset, the average result of the second dataset [32] on two folds is reported in Table 7. In this table, our proposed method's results are compared with the results of [19]. The evaluation results on the third [33] dataset are also reported in Table 7. The results of the method in [18] are compared with the results of our proposed method. Since the classifier's output probability on these two datasets is high, changing the threshold does not affect the results.

Based on the results shown in Table 7, our proposed method's accuracy for these datasets is better than state of the art solutions.

6. Conclusion

COVID-19 is a new infectious disease that spreads all over the world. Due to the highly contagious nature of this disease, its automatic detection is highly demanded to prevent its spread. Collecting a large number of COVID-19 samples in this crisis is a slow process. Therefore, training deep networks is prone to overfitting. However, computer vision approaches, like the bag-of-features technique, do not need a

Table 5
Results produced by different classifiers.

Classifier type	Accuracy	Sensitivity	Specificity
LogisticRegression	0.939	0.838	0.955
LinearSVC	0.944	0.806	0.965
KNeighborsClassifier	0.913	0.451	0.985
LinearDiscriminantAnalysis	0.935	0.581	0.99
GaussianNB	0.908	0.484	0.965
DecisionTreeClassifier	0.757	0.419	0.81
RandomForestClassifier	0.922	0.516	0.985
FCN	0.957	0.903	0.965

Table 6
The details of method for some papers in this field.

paper	#COVID test	#non COVID test	Feature Extraction	Classifier	#Class	Best ACC
[2]	25	25	ResNet50	SVM	2	95.38
[3]	–	–	COVID-CAPS	FCN	2	95.7
[6]	100	200	COVID_Net	FCN	3	93.3
[10]	25	50	Pre-trained CNN Models	SVM	3	95.33
[10]	25	50	GLCM, HOG, LBP,	SVM	3	93.5
[11]	68	560	ResNet	FCN	2[(1)]	96.1
[11]	68	560	InceptionV3	FCN	2[(1)]	95.4
[12]	162	–	Truncated Inception Net	FCN	2	98.77
[13]	864[(2)]	2686[(2)]	InceptionV3	FCN	3	96
[14]	25	200	Xception	FCN	3	97.4
[15]	455	3450	MobileNetV2	FCN	2	99.1
[16]	20	20	VGG16, VGG19, ResNet, DenseNet, InceptionV3	FCN	2	80
[17]	30[(3)]	–	Deep CNN	FCN	2	93
[18]	100	3000	ResNet18, ResNet50, SqueezeNet, and DenseNet-121	FCN	2	94
[19]	44	537	AlexNet, GoogLeNet, SqueezeNet	FCN	2	99.85
Our	31	200	Bag of Visual Words	SVM, FCN,	2	96.1
Our	571	1021	Bag of Visual Words	SVM, FCN,	2	99.84
Our	100	3000	Bag of Visual Words	SVM, FCN,	2	98

[(1)] The article only separates COVID-19 samples from normal samples not all of the **Pneumonia samples**.

[(2)] The articles have not mentioned the percentage of test and train images, so the whole number of images is reported.

[(3)] The article reported number of patients not the number of images.

Table 7
Results of methods in [2,3,18,19] and the proposed method.

Dataset	Approach	Accuracy	Sensitivity	Specificity
[31]	[2]	95.38	97.29	93.47
	[3]	95.7	90	95.8
	Proposed method	96.1	87.09	97.5
[32]	[19]	99.22	99.14	99.26
	Proposed method	99.84	99.65	99.95
[33]	[18]	92.29	98	92.9
	Proposed method	98	91	98.23

large dataset. In this paper, a classifying method based on a bag of features is proposed.

The three main phases of this approach are preprocessing, creating a dictionary, and classifying the images. In our proposed method, features are extracted with a handcrafted technique, and they are fed to a classifier neural network. The proposed method is compared experimentally with the state of the art methods. As illustrated in Table 7, our proposed method's accuracy on three datasets is 96.1, 99.84, and 98, respectively. These results are better than the state of the art methods. On the first dataset, our proposed method's sensitivity is susceptible to the threshold, and with the default threshold, it is less than state of the art.

The SURF algorithm is used in the bag of visual words for feature extraction. The SURF algorithm is gradient-based, and hence, the proposed method can be susceptible to noise. If the image's quality is low, the proposed method can misclassify images. In this situation, selecting an appropriate preprocess method can improve the results. For this purpose, intensity improvement and histogram matching methods are applied to images.

CRediT authorship contribution statement

Zahra Nabizadeh-Shahre-Babak: Methodology, Software, Investigation, Writing - original draft. **Nader Karimi:** Formal analysis, Validation. **Pejman Khadivi:** Methodology, Validation, Writing - review & editing. **Roshanak Roshandel:** Methodology, Writing - review & editing. **Ali Emami:** Software, Validation. **Shadrokh Samavi:** Conceptualization, Writing - review & editing, Supervision.

Declaration of Competing Interest

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 paper.

References

[1] Saima Hamid, Mohammad Yaseen Mir, Gulab Khan Rohela, Noval coronavirus disease (COVID-19): a pandemic (Epidemiology, pathogenesis and potential therapeutics), New Microbes New Infect. (2020), 100679.
[2] Prabira Kumar Sethy, Santi Kumari Behera, Detection of Coronavirus Disease (COVID-19) Based on Deep Features, 2020. Preprints 2020030300: 2020, https://www.preprints.org/manuscript/202003.0300/v1.
[3] Parnian Afshar, Shahin Heidarian, Farnoosh Naderkhani, Anastasia Oikonomou, Konstantinos N. Plataniotis, Arash Mohammadi, COVID-caps: a capsule network-based framework for identification of COVID-19 cases from x-ray images, arXiv preprint arXiv:2004.02696 (2020).
[4] Chuansheng Zheng, Xianbo Deng, Qing Fu, Qiang Zhou, Jiapei Feng, Hui Ma, Wenyu Liu, Xinggang Wang, Deep learning-based detection for COVID-19 from chest CT using weak label, medRxiv (2020).
[5] Feng Shi, Jun Wang, Jun Shi, Ziyan Wu, Qian Wang, Zhenyu Tang, Kelei He, Yinghuan Shi, Dinggang Shen, Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19, arXiv preprint arXiv:2004.02731 (2020).
[6] Linda Wang, Zhong Qiu Lin, Alexander Wong, COVID-net: a tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images, Sci. Rep. 10 (1) (2020) 1–12.
[7] Shuai Wang, Bo Kang, Jinlu Ma, Xianjun Zeng, Mingming Xiao, Jia Guo, Mengjiao Cai, et al., A deep learning algorithm using CT images to screen for corona virus disease (COVID-19), medRxiv (2020).
[8] Ying Song, Shuangjia Zheng, Liang Li, Xiang Zhang, Xiaodong Zhang, Ziwang Huang, Jianwen Chen, et al., Deep learning enables accurate diagnosis of novel coronavirus (COVID-19) with CT images, medRxiv (2020).
[9] Mucahid Barstugan, Umut Ozkaya, Saban Ozturk, Coronavirus (COVID-19) classification using CT images by machine learning methods, arXiv preprint arXiv:2003.09424 (2020).
[10] Prabira Kumar Sethy, Santi Kumari Behera, Pradyumna Kumar Ratha, Preesat Biswas, Detection of coronavirus disease (COVID-19) based on deep features and support vector machine, Int. J. Math., Eng. Manage. Sci. (2020).
[11] Ali Narin, Ceren Kaya, Ziynet Pamuk, Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks, arXiv preprint arXiv:2003.10849 (2020).
[12] Dipayan Das, K.C. Santosh, Umapada Pal, Truncated inception net: COVID-19 outbreak screening using chest X-rays, Phys. Eng. Sci. Med. 43 (3) (2020) 915–925.
[13] Sohaib Asif, Yi Wenhui, Automatic detection of COVID-19 using X-ray images with deep convolutional neural networks and machine learning, medRxiv (2020).

[14] N.Narayan Das, Naresh Kumar, Manjit Kaur, Vijay Kumar, Dilbag Singh, Automated deep transfer learning-based approach for detection of COVID-19 infection in chest X-rays, Irbm (2020).

[15] Ioannis D. Apostolopoulos, Sokratis I. Aznaouridis, Mpesiana A. Tzani, Extracting possibly representative COVID-19 biomarkers from X-ray images with deep learning approach and image data related to pulmonary diseases, J. Med. Biol. Eng. (2020) 1.

[16] K. Sahinbas, F.O. Catak, Transfer Learning Based Convolutional Neural Network for COVID-19 Detection With X-Ray Images, 2020. https://www.ozgurcatak.org/files/papers/covid19-deep-learning.pdf.

[17] Md Jamil, Iftekhar Hussain, Automatic detection of COVID-19 infection from chest X-ray using deep learning, medRxiv (2020).

[18] Shervin Minaee, Rahele Kafieh, Milan Sonka, Shakib Yazdani, Ghazaleh Jamalipour Soufi, Deep-Covid: predicting covid-19 from chest x-ray images using deep transfer learning, arXiv preprint arXiv:2004.09363 (2020).

[19] Tuan D. Pham, Classification of COVID-19 chest X-rays with deep learning: new models or fine tuning? Health Inf. Sci. Syst. 9 (1) (2021) 1–11.

[20] Qingsen Yan, Bo Wang, Dong Gong, Chuan Luo, Wei Zhao, Jianhu Shen, Qinfeng Shi, Shuo Jin, Liang Zhang, Zheng You, COVID-19 chest CT image segmentation—A deep convolutional neural network solution, arXiv preprint arXiv:2004.10987 (2020).

[21] Deng-Ping Fan, Tao Zhou, Ge-Peng Ji, Yi Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, Ling Shao, Inf-Net: automatic COVID-19 lung infection segmentation from CT scans, arXiv preprint arXiv:2004.14133 (2020).

[22] Amine Amyar, Romain Modzelewski, Su Ruan, Multi-task deep learning based CT imaging analysis for COVID-19: classification and segmentation, medRxiv (2020).

[23] Liangping Tu, Changqing Dong, Histogram equalization and image feature matching, in: 2013 6th International Congress on Image and Signal Processing (CISP), vol. 1, IEEE, 2013, pp. 443–447.

[24] Rafael C. Gonzalez, Richard Eugene Woods, Steven L. Eddins, Digital Image Processing Using MATLAB, Tata McGraw-Hill Education, 2013.

[25] Stephen O'Hara, Bruce A. Draper, Introduction to the bag of features paradigm for image classification and retrieval, arXiv preprint arXiv:1101.3354 (2011).

[26] Andrew J. Newell, Lewis D. Griffin, Multiscale histogram of oriented gradient descriptors for robust character recognition, in: 2011 International Conference on Document Analysis and Recognition, IEEE, 2011, pp. 1085–1089.

[27] Herbert Bay, Tinne Tuytelaars, Luc Van Gool, Surf: speeded up robust features, in: European Conference on Computer Vision, Springer, Berlin, Heidelberg, 2006, pp. 404–417.

[28] Christopher G. Harris, Mike Stephens, A combined corner and edge detector, in: Alvey Vision Conference, vol. 15, 1988, pp. 10–5244, 50.

[29] David G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[30] John A. Hartigan, Manchek A. Wong, Algorithm AS 136: a k-means clustering algorithm, J. R. Stat. Soc. Ser. C Appl. Stat. 28 (1) (1979) 100–108.

[31] https://github.com/lindawangg/COVID-Net/blob/master/docs/COVIDx.md.

[32] https://www.kaggle.com/tawsifurrahman/covid19-radiography-database.

[33] https://github.com/shervinmin/DeepCovid/tree/master/data.

[34] Pejman Khadivi, Ravi Tandon, Naren Ramakrishnan, Flow of information in feed-forward denoising neural networks, Proceedings of IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (2018) 166–173.