A fast lightweight network for the discrimination of COVID-19 and pulmonary diseases

Oussama Aiadi [*], Belal Khaldi

Artifcial Intelligence and Information Technology Laboratory (LINATI), Department of Computer Science and Information Technology, Faculty of Sciences and Technology, University of Kasdi Merbah, Ouargla 3000, Algeria

ARTICLE INFO

ABSTRACT

With the outbreak of COVID-19 and the increasing number of infections worldwide, there has been a noticeable deficiency in healthcare provided by medical professionals. To cope with this situation, computational methods can be used in different steps of COVID-19 handling. The first step is to accurately and rapidly diagnose infected persons, because the time taken for the diagnosis is among the crucial factors to save human lives. This paper proposes a computationally fast network for the diagnosis of COVID-19 and pulmonary diseases, which can be used in telemedicine. The proposed network is called DLNet because it jointly encodes local binary patterns along with filter outputs of discrete cosine transform (DCT). The first layer in DLNet is the convolution layer in which the input image is convolved using DCT filters. Then, to avoid over-fitting, a binary hashing procedure is performed by fusing responses of different filters into a unique feature map. This map is used to generate block-wise histograms by binding local binary codes of the input image and the map values. We normalize these histograms to improve the robustness of the network against illumination changes. Experiments conducted on a public dataset demonstrate the rapidity and effectiveness of DLNet, where an average accuracy, sensitivity, and specificity of 98.86%, 98.06, and 99.24% have been achieved, respectively. Moreover, the proposed network has shown high tolerance to the missing parts in the medical image, which makes it suitable for the telemedicine scenario.

1. Introduction

Within a short period, COVID-19 has proliferated across the world, causing an increasing number of deaths as well as significant economic losses. According to [1], until January 2022, the number of confirmed cases is estimated at around 298 million along with more than 5 million deaths. In addition, new variants (i.e., Delta, Omicron …) of COVID-19 appear periodically, making it difficult to definitively get rid of this disease. As for the economic effects of this pandemic, there has been a significant rise in unemployment, perturbation in transportation chains, decrease in revenue, with fatal losses for the industrial section.

The reverse transcription-polymerase chain reaction (RT-PCR) technique is currently used for COVID-19 screening. However, RT-PCR takes a considerable deal of time to produce decisions with a high false-negative rate [2–5]. In addition, to take samples for RT-PCR, well-trained employers are required. Computed tomography (CT) and chest X-Ray (CXR) images, which are cheaper than RT-PRC, are also used for COVID-19 diagnosis. During peak times of COVID-19, in certain countries, RT-PCR becomes scarce and may be unavailable due to the perturbations of supply chains and the high demand for it, which increases its price. On the other hand, CT scan and CXR are already widely available and used in clinics for the diagnosis of different diseases before the appearance of COVID-19. Thus, the RT-PCR test standard for screening COVID-19 is expensive, time-consuming, and requires well-trained employers, whereas the CXR/CT images provide a time/cost-effective method for COVID-19 diagnosis. Nevertheless, compared to CXR, CT scan is considered more expensive and harmful to the patient than the former one.

With the huge proliferation of COVID-19 all over the world, medical personnel has been greatly affected by COVID-19, and there has been a noticeable number of infections among this personnel. This, along with the exponential increase in the number of patients everyday, has caused a remarkable shortage in healthcare provided by the medical staff. Thus, many countries have opted for artificial intelligence techniques to help fight this disease. In particular, deep learning has been widely adopted in several studies for COVID-19 detection from CXR images. It can be

---

* Corresponding author.
E-mail address: aiadi.oussama@univ-ouargla.dz (O. Aiadi).

noted that promising outcomes have been achieved by those studies, which encourages further investigations on this track. From one point of view, depending on the method used for detection, existing work can broadly be classified into two approaches, namely segmentation and feature learning. The first category, including the methods [6–9], aims to detect infected parts of the lung using segmentation techniques. The second approach [10–13] focuses on extracting features that can reflect the image content accurately. From another point of view, depending on the classification outcomes, current works can be categorized into two categories. The first category of works performs a binary classification i. e., classify patients into COVID-19 and non-COVID-19, whereas, the second category includes a third class and considers pulmonary diseases such as viral pneumonia and fibrosis.

Although a considerable number of methods have been proposed during the last two years, there is still a great deal of room for improvement. For instance, most existing works consider using supervised deep-based schemes that often require a significant deal of time for training. In addition, the processing time required to make a prediction is proportional to the number of layers/parameters. Time is a quite crucial factor in the diagnosis of COVID-19 and plays a decisive role in saving human lives. Along with the problem of high computation and data storage, the main disadvantage of the CNN-based approaches is the dependency on a training dataset that should be large enough to reach a satisfactory generalization power. While the performance achieved by the previous works is promising, it remains very challenging to complete the other desirable aspects along with the good performance. It is challenging to design a method capable to deliver precise predictions of COVID-19 in a reasonable time, with limited computational resources and high robustness against degradation of the input images.

Indeed, during peak times of infections, COVID-19 diagnosis via computationally expensive methods will not beneficial for breaking the chain of infection. Moreover, with the remarkable advances in the field of telemedicine, it becomes possible for medical professionals to make remote diagnoses and monitor. Teleradiology is a typical task of telemedicine, which allows radiologists to make remote interpretations of medical images. In the COVID-19 pandemic, with the deficiency of medical health practitioners, such a process can help break the chain of infection and save human lives. In such a scenario, transferred medical images may have missing parts because of transmission errors. In such a case, patients may resort to retaking another clearer image. Therefore, using methods with minor robustness may cause the diagnosis process to delay and can negatively affect the decisions reached by the radiologists. Furthermore, adopting methods with high computation and storage requirements will incur extra costs on the financial budget devoted to fighting COVID-19. Addressing the above-mentioned problems will have a direct impact on limiting the proliferation of COVID-19, facilitating the diagnosis process, reducing the required computational cost, and thus, saving human lives all over the world.

In this paper, we propose a lightweight network for COVID-19 and pulmonary disease recognition from CXR images. The main objective of this study is to make this network effective, efficient, robust, and computationally fast. We refer to this network as DLNet because it fuses feature maps produced by the convolution layer with local binary codes associated with pixels in the original CXR image. The novelty of DLNet lies in its:

- Simple architecture: it is composed of a single convolution layer with a few stacked steps.
- Effectiveness: this is due to the incorporation of filter responses from the convolution layer with local binary patterns associated with different pixels. In addition, considering the binary hashing process allows for avoiding over-fitting and improving the generalization power of the network. Furthermore, normalizing learned features improves the network robustness against illumination changes.
- Low computational cost and efficiency: due to its simple architecture in which features learning is performed without the need for an

intensive training process. In addition, instead of data-driven filters, data-independent filters are used in the single convolution layer, which reduces the method cost and improves its efficiency. Moreover, it can achieve high performance with limited resources (i.e., in terms of computation and storage).
- Robustness to the missing parts in the medical images: this makes it suitable for telemedicine scenarios due to the block-wise manner used for feature learning and normalization.

We conduct thorough experiments to evaluate the proposed DLNet on a public dataset that is made up of 4000 CXR images from three different classes namely COVID-19, pulmonary diseases, and healthy persons. As for pulmonary diseases, 19 kinds of diseases are considered involving fibrosis, SARS, pneumonia, atelectasis, and cardiomegaly. The dataset is gathered, during the last two years, from different hospitals and medical associations around the world including the Italian Society of Medical and Interventional Radiology and hospitals in China, Italy, USA, Australia, Korea Taiwan, and Sweden. It is worth noting that we mimic the scenario of telemedicine by successively cropping distinct regions from the medical image. Experimental results demonstrate the effectiveness of the proposed network, where an average accuracy, sensitivity, and specificity of 98.86%, 98.06, and 99.24% have been achieved, respectively.

The remainder of this paper is organized as follows. Section 2 provides an overview of the works concerned with COVID-19 recognition. Section 3 presents our proposed DLNet. Section 4 reports the experimental results. Finally, Section 5 presents the work conclusions and the future directions.

2. Related work

In this section, we review related works concerned with COVID-19 diagnosis. At first, we present studies focusing on feature learning, followed by studies that have adopted segmentation for COVID-19 screening.

2.1. Feature learning approaches

Determining the feature that is capable of faithfully describing the image content is the cornerstone of recognition systems. Some researchers have opted for handcrafted features, such as gray-level co-occurrence matrix (GLCM) and histogram of gradients (HoG) [14–16], while, most researchers have considered using different deep architectures to automatically learn effective representations from chest images. For instance, in [16], GLCM and two other texture features were used to detect COVID-19 from CXR images. In another work [15], an experimental assessment was carried out to measure the performance of different texture features, including deep and handcrafted ones. Similarly, authors in [14] considered employing HoG together with CNN for the diagnosis of COVID-19 and pneumonia.

Although handcrafted features are not data hungry and they have achieved promising results, their generalization power is minor compared to deep-based architectures. Additionally, handcrafted features are sensitive to the highly similar classes, which require the scientist to make amendments to the feature design to be capable to distinguish those classes.

The work in [13] suggests combining features extracted using two pre-trained CNNs namely ShuffleNet and SqueezeNet. These features are then fed to a multi-class SVM to recognize three types of diseases which are COVID-19, bacterial pneumonia, and viral pneumonia. In [10], two customized CNN architectures (CovidResNet and CovidDenseNet) were proposed to detect COVID-19 from chest CT images. The two models can be partly initialized with larger networks such as ResNet50. Authors of [3] have proposed a two-stage scheme to discriminate pneumonia and COVID-19 using deep learning. The first stage is dedicated to check if there is a pneumonia, whereas, COVID-19 and pneumonia are
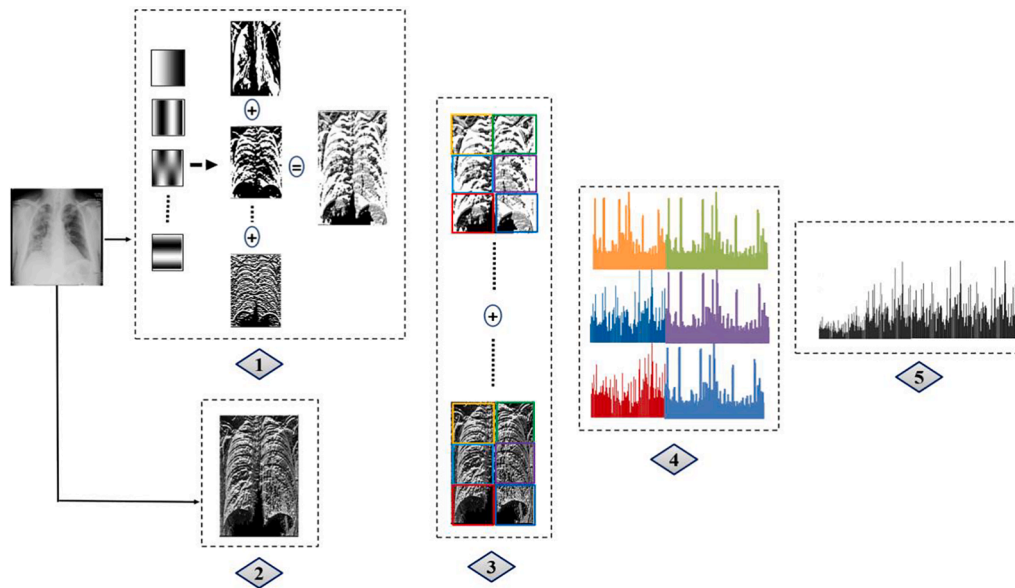
Fig. 1. The general architecture of the proposed network: 1) convolution layer and binary hashing, 2) calculation of LBP image, 3) block-wise histograms generation and normalization and 5) the final histogram.

discriminated in the second stage. In [17], authors proposed a method for COVID-19 diagnosis from CT scan images, where the two-dimensional fractional Fourier entropy is used for features extraction. Note that along with COVID-19, three other classes namely community-acquired pneumonia, secondary pulmonary tuberculosis, and healthy control are considered. To classify test images, a custom deep stacked sparse autoencoder is created. In addition, the authors proposed an improved multiple-way data augmentation technique to avoid overfitting.

In [18], an *n*-conv rank-based average pooling module (NRAPM) was proposed in which rank-based average pooling was employed to prevent overfitting. Then, inspired by the VGG network and with NRAPM-based conv blocks, a deep rank-based average pooling (DRAPNet) was proposed. Noting that a custom improved multiple-way data augmentation procedure was firstly performed. For the sake of explainability, the authors used the Grad-CAM method to generate the heatmaps. Experiments were conducted on a dataset of 521 subjects yielding 1164 slice images via the slice level selection method. This dataset is composed of four classes which are COVID-19 positive, community-acquired pneumonia, second pulmonary tuberculosis, and healthy control. This method has achieved a micro-averaged F1 score of 95.49%.

Some other studies have investigated different kinds of fusion schemes. For instance, in [4], a graph convolutional network was used together with CNN to detect COVID-19. To enhance the diagnosis outputs, a deep convolutional attention network with multiple inputs was used to fuse chest X-Ray and CT images in [19]. Likewise, to strengthen decisions reached by individual networks, authors of [12] used three pre-trained VGG-16, where inputs of the networks are original CXR and HSV image along with a third image processed using Prewitt operator. Another form of fusion is considered in [20], where a total of 7 pre-trained CNNs were fused at decision and feature level to detect COVID-19. In [21], a stacked auto-encoder (4 auto-encoders) model was developed to improve COVID-19 detection from CT images. Nevertheless, CNN strongly depends on the training dataset and suffers from the issue of high computation and data storage. This dataset should be large enough to ensure the generalization power of the network. In addition, in the case of designing a CNN from scratch, it becomes more complicated due to the little theoretical guidance. Furthermore, picking out the appropriate CNN architecture is challenging because the number and nature of layers may vary depending on the problem being solved. As for

methods considering the fusion of different CNNs, such kinds of fusions incur additional computational costs to these deep-based methods which are, inherently, computationally intensive.

2.2. Segmentation-based approaches

Segmentation-based approaches aim to detect infected regions from chest images by using segmentation methods. For instance, Zernike moments and GLCM were used with a deep neural network to localize infected regions from CT images in [7]. The very well-known U-Net architecture was the base of the work in [22], where a dilated dual attention U-Net is proposed to segment COVID-19 lesions. Similarly, U-Net was considered in [23], where an extensive data augmentation procedure was followed to prevent the network from overfitting. Authors in [8] proposed a multi-scale discriminative network for segmenting COVID-19, which incorporates three blocks namely pyramid convolution, channel attention and residual refinement. This last is designed to boost segmentation results by considering kernels of different sizes. In [24], authors have proposed an encoder-decoder deep architecture to determine the infected regions within the lung. On the one hand, the encoder part is composed of multiple layers namely convolution, batch normalization, RELU, and max pooling. On the other hand, the decoder part is composed of upsampling, convolution, batch normalization, and RELU. In the proposed architecture, instead of using one encoder, two encoders are employed, where the final encoder feature map is formed by concatenating the output of the two encoders. The first stage aims to segment the region of interest. Thus, the input encoders receive two images namely texture and structural components of the input image, and produce two feature maps. These maps are concatenated and fed to the decoder which produces an image containing the region of interest. The second stage aims to segment infected lung regions. To do so, two encoders receive the region of interest image (produced by the first stage) and the input image, and output two feature maps. These maps are concatenated and fed to the decoder which generates an image containing the infected regions.

To deal with the data scarcity issue, an unsupervised domain adaptation-based segmentation network was proposed in [25]. Two types of data namely synthetic data and limited unlabeled CT images of COVID-19 were used to train this network. Authors in [26] have coped with this issue differently by using an improved dense generative
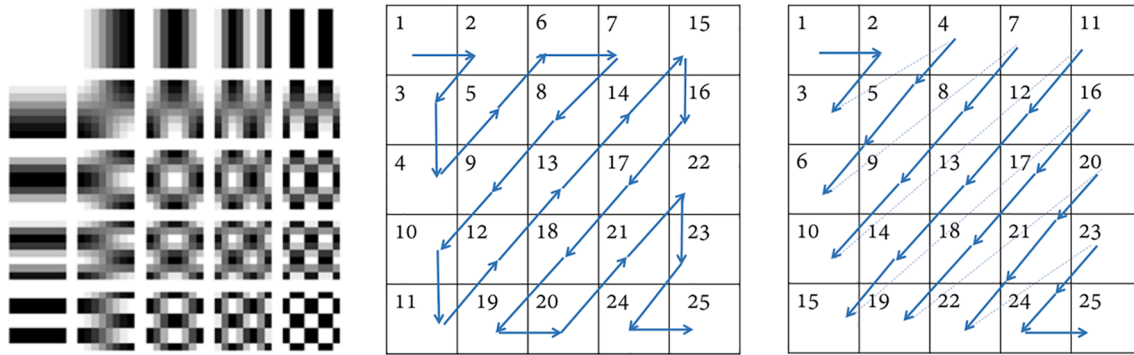
Fig. 2. The DCT filter bank (the leftmost image) and the two strategies for filter selection: the zig-zag (in middle) and the horizontal-frequency major ordering (the rightmost image).

adversarial network (GAN) to expand the existing dataset of COVID-19 images. However, it remains very challenging to achieve accurate segmentation of COVID-19 for several reasons. First, different types of infected regions have distinct physical appearances. Secondly, infected areas may greatly vary in terms of size, shape, location, and texture. This alongside the intensity inhomogeneity of the infected regions as well as the blurred boundaries between lesions and normal tissues makes it very difficult to precisely detect the infected areas.

3. Proposed method

As has already been mentioned, it is very challenging to develop an efficient, robust, and computationally fast network that can reach accurate classification outcomes and distinguish between patients infected by COVID-19, pulmonary diseases, and healthy persons. In this section, we present our proposed lightweight network (DLNet). Fig. 1 illustrates the general flowchart of DLNet.

ZDNet has a single convolution layer in which input medical images are convolved using data-independent DCT filters. This allows for reducing the computational cost and improving the network efficiency, as generating data-driven filters will take a significant deal of time, especially if the training is performed using a large-scale dataset. Local binary patterns of the input image are then calculated to characterize texture of the medical image. Then, to alleviate over-fitting, feature maps from the convolution layer are fused into a single map using binary hashing. The bottleneck of the proposed network lies in generating a histogram that jointly considers filter responses (from the final feature map) and LBP codes associated with local image pixels. Instead of generating separate histograms for each of LBP and the feature map, the generated histogram allows binding both histograms to strengthen the image representation. We consider incorporating spatial relationships by extracting the histograms from local image regions. Then, extracted histograms are normalized against illumination changes. Finally, a support vector machine classifier is used for test image matching. Hereafter, we provide more details on each step in DLNet.

3.1. Convolutional filters generation

Assuming that we are given an input image, denoted by I with the dimensions $M \times N$. We consider using discrete cosine transform (DCT) filter bank, which allows our network to have the property of data-independency (i.e., in contrast to data-driven filters). Although they

are extracted differently, the equivalence of DCT and principal component analysis (PCA) filters has been proven in [27]. By extending 1D DCT, 2D DCT is given by.

$$f(u,v) = \delta(u)\delta(v) \sum_{i=1}^{M} \sum_{j=1}^{N} f(x,y) \cos\left[\frac{\pi(2i+1)u}{2M}\right] \cos\left[\frac{\pi(2j+1)u}{2N}\right], \quad (1)$$

where

$$\delta(u) = \begin{cases} \dfrac{1}{\sqrt{M}}, & if u = 0 \\ \sqrt{\dfrac{2}{M}}, & otherwise \end{cases} \quad (2)$$

$$\delta(v) = \begin{cases} \dfrac{1}{\sqrt{N}}, & if v = 0 \\ \sqrt{\dfrac{2}{N}}, & otherwise \end{cases} \quad (3)$$

The filter bank is composed of several distinct 2D DCT bases. To rank the DCT filters and pick out the most eligible ones we can use with the convolution layer, we consider the horizontal-frequency major ordering instead of zig-zag ordering (see Fig. 2). Generally speaking, in DCT, coefficients of bases with low frequency are higher than coefficients of high-frequency bases because humans are more sensitive to low-frequency bases. In both kinds of ordering schemes, diagonal bases are sequentially ranked. However, the difference lies in the importance assigned to those bases. While in the zig-zag ordering, frequency direction importance is alternated between horizontal and vertical bases, the horizontal-frequency major ordering gives more importance to horizontal filters (i.e., they are prioritized over the vertical filters). This last selection strategy is consistent with the nature of images being processed in which low-frequency horizontal bases are more likely to occur. Fig. 3 shows some selected bases from the filter bank.

3.2. Convolution layer

The DCT filters generated in the previous step are used to perform convolution operations in the spatial domain. In particular, the DCT filters are used to convolve input medical images in the first layer of DLNet. Hereafter, we give more details on this process.

Suppose that the 2D filter size is $k \times k$, I is convolved with different 2D DCT bases as follows.
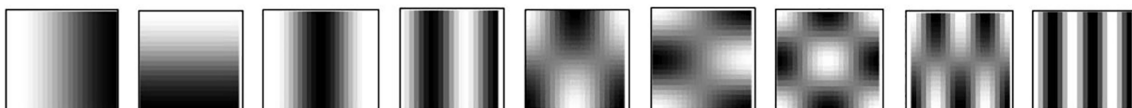
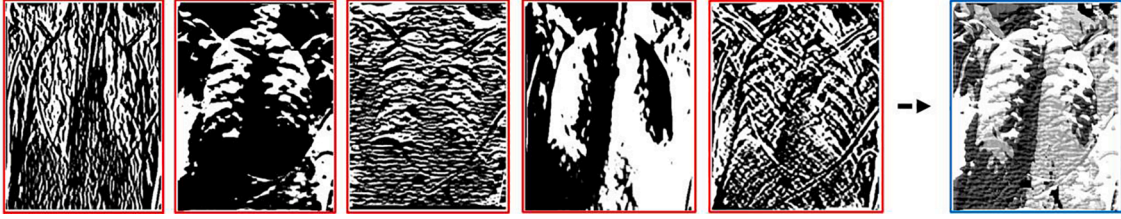

Fig. 3. Typical filters from the DCT filter bank.

Fig. 4. Binary hashing process, images with a red border are feature maps and the one with a blue border is the output of fusing those images.
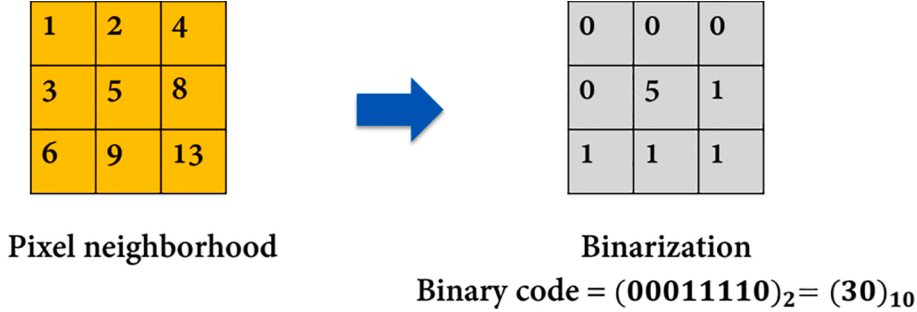

Pixel neighborhood

Binarization

Binary code = $(00011110)_2 = (30)_{10}$

Fig. 5. An example of LBP calculation.

$$O^p = \left\{I * W_p\right\}_{p=1}^{L} \tag{4}$$

Where $W_p \in \mathbb{R}^{k \times k}, p = 1, \cdots, L$ represents the set of 2D DCT bases (i.e., filters) and p_L stands for the number of filters. Noting that feature maps O^p has the same size as I because borders of image is zero padded with pad size of $(k-1)/2$ before performing convolution.

3.3. Binary hashing

To prevent our network from over-fitting, we consider performing a binary hashing procedure on feature maps by quantifying filter responses. After convolving the input image using DCT filters, we obtain the feature maps, where their values include real numbers. In this step, we binarize each map by considering zero as a threshold i.e., values higher than zero (i.e., positive) are replaced by one, and zero otherwise. Eq. (5) defines the binarization process. This step is preliminary to the next step (i.e., binary hashing) which aims to prevent over-fitting and characterize filter responses by accumulating positive values of feature maps.

$$BIN(O^p) = \begin{cases} 1, if\, O^p > 0 \\ 0, otherwise \end{cases} \tag{5}$$

The binarized feature maps are combined to form a single image denoted by D (Fig. 4). In this case, every pixel in D will range from 0 to 2^{L-1}. The Combination is done according to the next equation.

$$D = \sum_{p=1}^{L} 2^{p-1} BIN(O^p) \tag{6}$$

3.4. Image encoding using local binary patterns

In the previous steps, we have considered pixel responses to different filters. In this step, the aim is to strengthen our DLNet by including local binary patterns (LBP) associated with different image pixels. To do so, we calculate binary codes of pixels in the input image. In the basic LBP, a 3×3 neighborhood around each pixel is considered, then, pixels have a higher value than the center pixel are assigned by one, and zero otherwise. The binary code is obtained by taking the binarized values in a clockwise direction (see Fig. 5). Formally, for a pixel g_c defined by is

coordinates $I(x_c, y_c)$ and surrounded by a neighborhood of p pixels. LBP code is generated according to the following equation.

$$LBP_{g_c} = \sum_{i=0}^{i=p-1} BIN(g_p - g_c) 2^i \tag{7}$$

3.5. Block-wise filter response-based LBP histogram generation

The cornerstone of DLNet lies in jointly considering two kinds of crucial information which are filter responses as well as local binary codes of image pixels. To reach such a representation, we extract a 2^M-dimensional histogram that binds local binary patterns and filters responses. This histogram is referred to as *HISTL*, and can be generated using Eq. (8).

$$HISTL(w) = \sum_{a=1}^{u} LBP_{g_c}(Ind_w) \tag{8}$$

such that $u = |Ind_w|$ and w represents the index of the histogram. Ind_w is the set of pixels' spatial coordinates for which D is equal to w. Ind_w is defined by.

$$Ind_w \equiv (D = w) \tag{9}$$

where the operator $\equiv$ is an assignment operator that assigns Ind_w with the spatial coordinates for which $D = w$. To take advantage of the spatial relationship, we extract the histogram in a block-wise fashion. Therefore, D and LBP_{g_c} are divided into non-overlapping blocks, and histograms extracted from different blocks are concatenated in a single histogram, which is, to some extent, translation-invariant.

Indeed, there is a close relationship between the outputs of this step and the binary hashing process which aims to prevent overfitting [28,29]. To understand how can this be done, we give a simple example. Suppose that number of filters is equal to R. Thus, the output of the convolution layer will be R feature maps. Generating local histograms using all these feature maps along with the LBP image will result in a high-dimensional feature vector. For instance, if we have an $N \times N$ image, a block size of $M \times M$ pixels and k filters, the dimension of the local feature vector will be 2^k. We have a total of $\left(\frac{M}{N}\right)^2$ blocks in the entire image, thus, dimension of the final feature vector will be $2^k \times \left(\frac{M}{N}\right)^2$. However, if binary hashing is excluded, dimension of the final
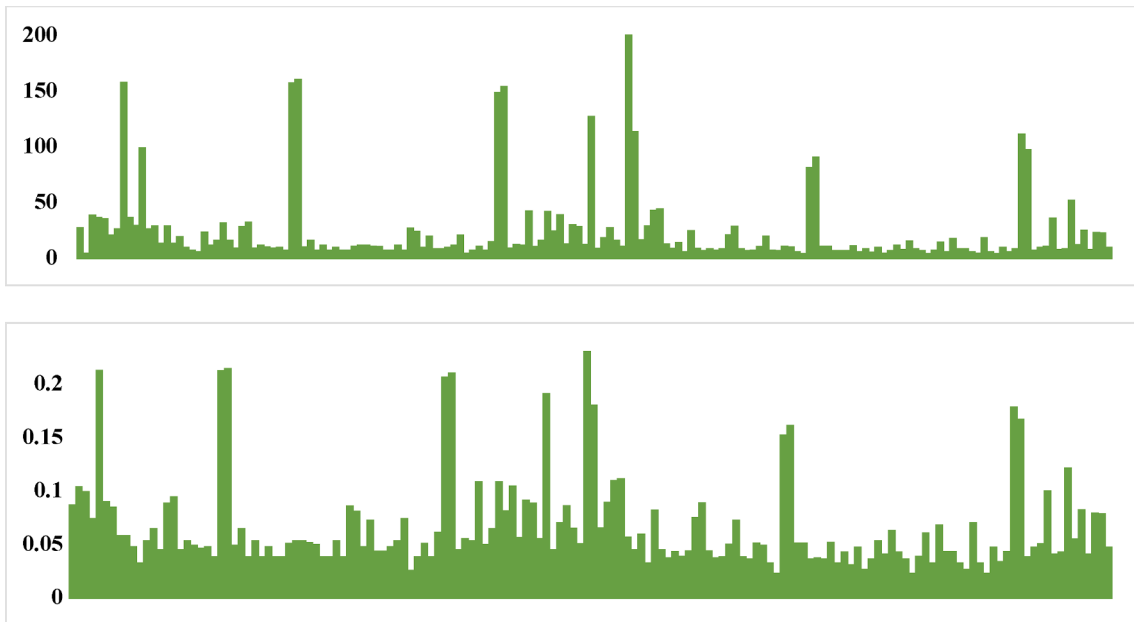
Fig. 6. Histogram prior to normalization (at top) and histogram after normalization (at the bottom). This latter seems to be more evenly distributed.

Table 1
list of the pulmonary diseases considered by this study.

Disease	Number	Disease	Number	Disease	Number	Disease	Number
ARDS	4	Chlamydophila	2	Escherichia	4	Hernia	7
Klebsiella	1	Legionella	2	Pneumocystis	14	Infiltration	623
SARS	15	Streptococcus	16	Atelectasis	508	Pneumonia	500
Cardiomegaly	125	Consolidation	160	Edema	93	Mass	147
Effusion	392	Emphysema	84	Fibrosis	56		

feature vector will be $2^k \times \left(\frac{M}{N}\right)^2 \times R$, which can significantly increased in the case of high-resolution images. This high dimension will cause the network to strictly fit the training images, thus, reducing the generalization power of the network. In addition, high dimensional feature vectors can greatly affect the network efficiency. To avoid such issues, we consider fusing all the feature maps into a single one using the binary

hashing according to Eq. (5) and Eq. (6).

3.6. Histogram normalization and matching

We perform histogram normalization to improve the robustness against illumination changes. In fact, the significant disparity in features
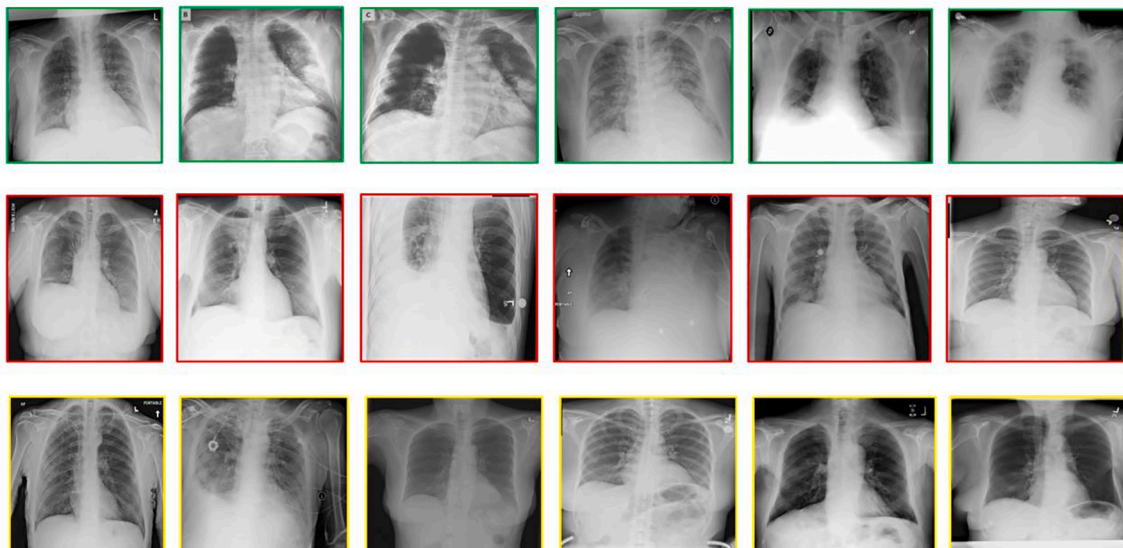

Fig. 7. Samples from the used dataset. The first row corresponds to healthy samples, the second row corresponds to pulmonary diseases (from left to right: fibrosis, emphysema, effusion, atelectasis, pneumonia and Cardiomegaly), the last row corresponds to COVID-19 samples.

Table 2
Different parameters subsets that are experimented.

Subset of parameters	Number of filters	Filter size	Block size
1	5	5×5	50×50
2	5	7×7	200×200
3	8	9×9	25×25
4	9	9×9	100×100
5	9	9×9	50×50
6	7	7×7	100×100

can noticeably degrade the classification results. Thus, a power-L_2 normalization scheme is considered to relieve this disparity. Given a histogram $H = (h_1, \cdots, h_f)$, power-L_2 is defined as follows.

$$h_i = \frac{\sqrt[2]{h_i}}{\|H\|} \tag{10}$$

$\|H\|$ stands for the L_2 norm of H. Fig. 6 depicts the effect of normalization. In this work, a linear one versus all support vector machine (SVM) classifier is used for histogram matching.

4. Experiments and discussion

4.1. Dataset

To assess the performance of the proposed method, we consider creating a dataset of CXR images from two different datasets [30,31]

which are publicly available at [32,33]. This dataset is made up of 4000 CXR images. More specifically, images are divided into three different classes: 1500 images for patients infected by COVID-19, 1250 images related to pulmonary diseases and the remaining images (i.e., 1250) are for healthy cases. Table 1 presents the list of pulmonary diseases included in the dataset. Fig. 7 shows representative images from our dataset. 1000 images from each class are intended for training and the rest are for testing.

4.2. Performance metrics

Three measures were utilized to evaluate the classification performance of our DLNet namely accuracy, specificity and sensitivity. They are defined as follows.

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

$$specificity = \frac{TN}{FP + TN}$$

$$sensitivity = \frac{TP}{FN + TP}$$

Where *TP* stands for the number of true positives, *TN* represents the number of true negatives, *FN* is the number of false negatives and *FP* is the number of false positives.
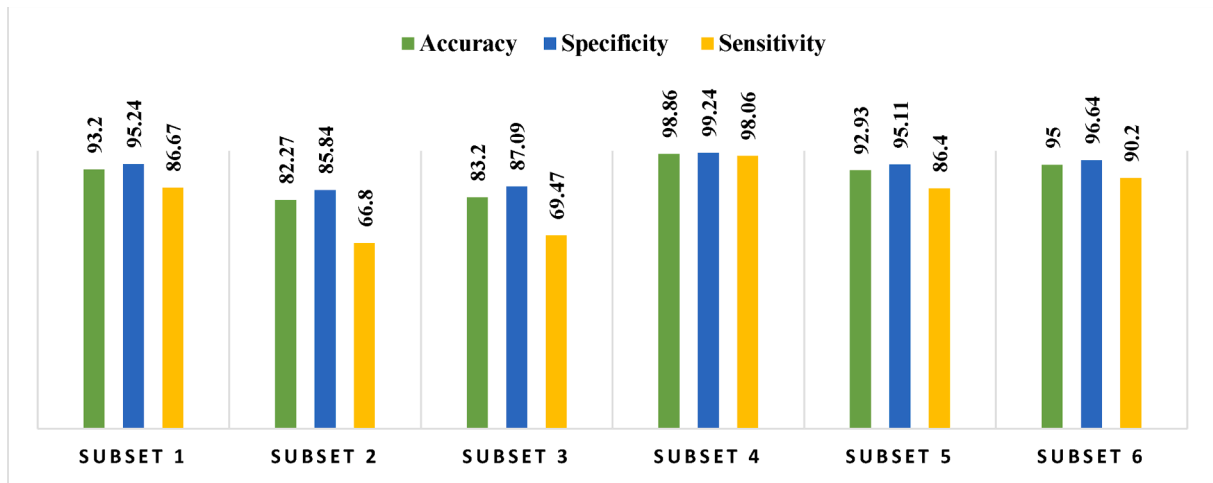

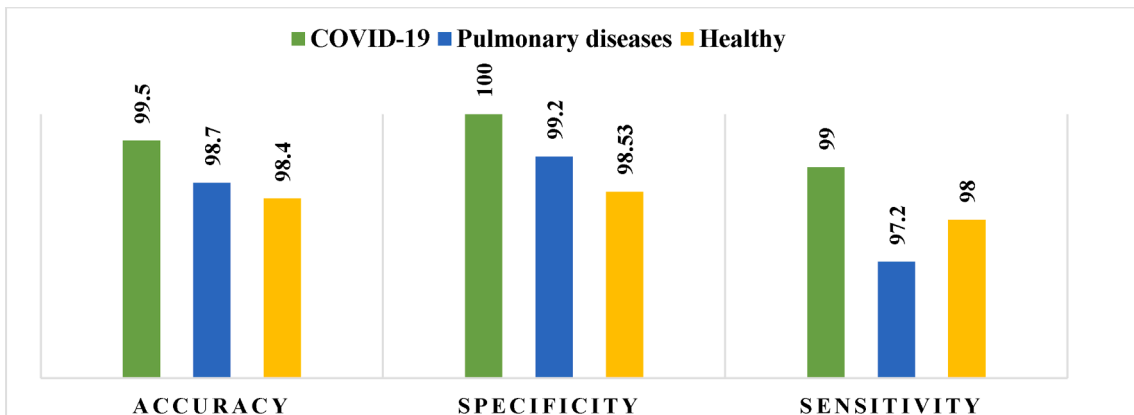**Fig. 8.** Performance achieved by different subsets.

Fig. 9. Performance per class reached by DLNet.

COVID-19   Healthy   Pulmonary

	COVID-19	Healthy	Pulmonary
COVID-19	495	4	1
Healthy	0	245	5
Pulmonary	0	7	243

Actual class   Predicted class

Fig. 10. Confusion matrix of the proposed network.

4.3. Dlnet parameters tuning

In this experiment, we measure the performance of the proposed method when varying different parameters. We do so to detect the subset of parameters that can improve the classification results. DLNet has three hyper-parameters which are the number of filters, filter size and block size. Among several, we have tested six subsets as shown in Table 2.

Fig. 8 reports the classification results for the different subsets in Table 2. From this figure, we can make the following remarks.

- The first remark we can make is the relative stability of classification scores for most subsets despite the change in parameters. Another thing to note is that specificity is higher than sensitivity for the six subsets, which means that, for the three classes, the number of true negatives is higher than the number of true positives.

- If we look at the number of filters, we can notice that this parameter can noticeably affect the classification outcomes. In our case, the maximum performance is achieved by using 9 filters. Nevertheless, it should be noted that the number of filters is proportional to the feature dimensions. Therefore, this parameter has to be a compromise between feature dimensions and classification accuracy.
- Generally speaking, filter size must be large enough to describe the interesting regions within the chest image. Small size filters will cause the network to over-fit because they will significantly increase the feature dimensions. In this experiment, the 9×9 filter has scored the best result.
- As for the block size, we can see that a 100×100 block is suitable for our dataset.
- Finally, by comparing all the subsets, we can note that the fourth subset has yielded the best recognition scores (accuracy = 98.86%, specificity = 99.24%, and sensitivity 98.06%).
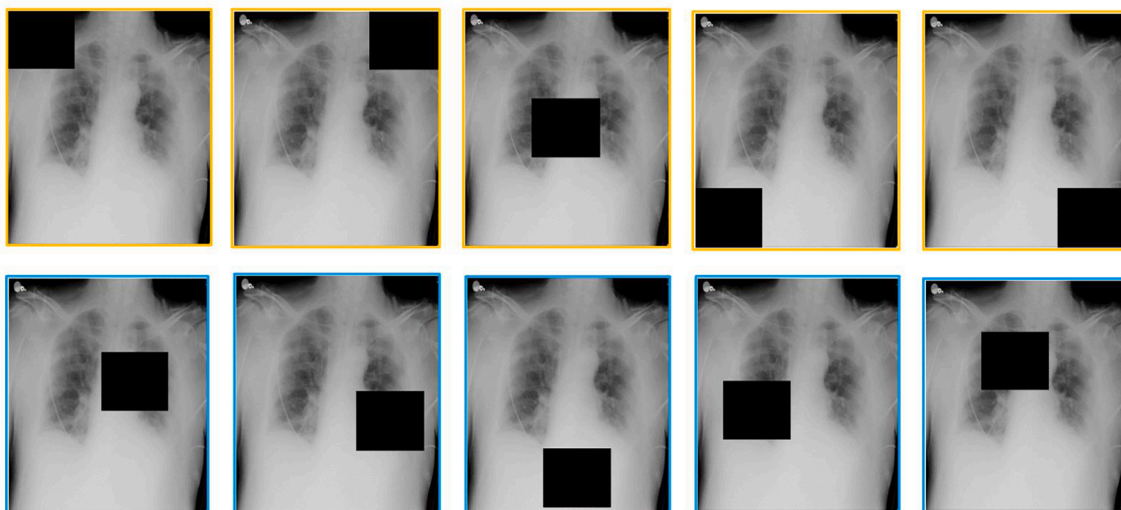

Fig. 11. Settings adopted to test the tolerance to missing parts in the image, top row corresponds to the first five settings which are referred to as (from left to right) S1, S2, S3, S4 and S4. The second row depicts the random settings which are denoted by (from left to right): RS1, RS2, RS3, RS4 and RS5.
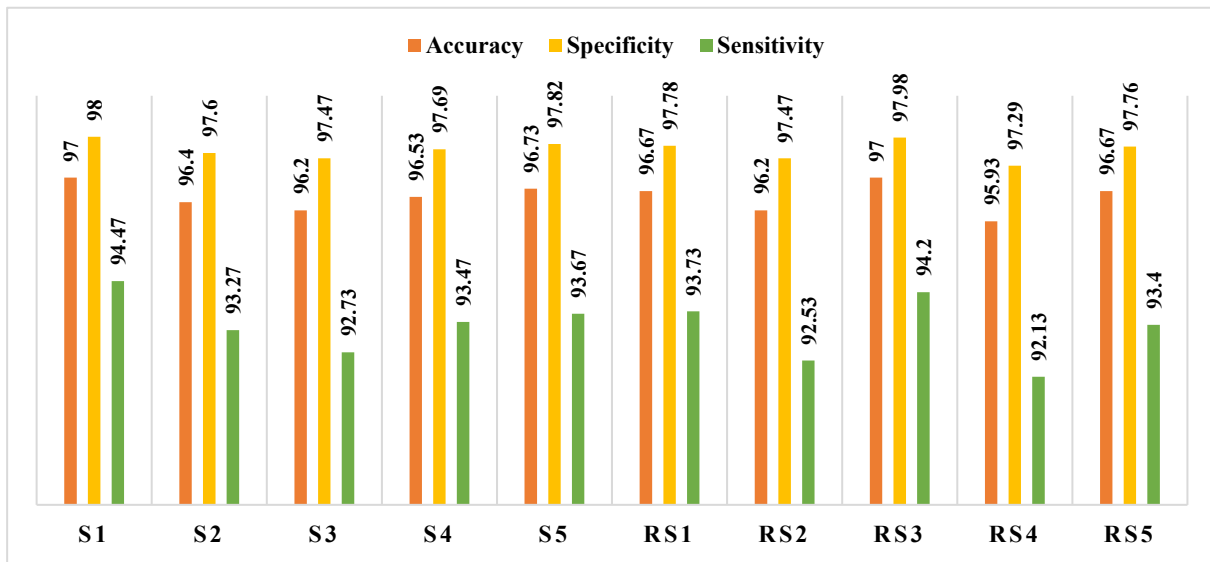
Fig. 12. Classification scores yielded by different settings.

For a comprehensive analysis, we report the classification scores per class (Fig. 9). We can see that 99.5% (i.e., about 495 out of 500) of COVID-19 images are correctly classified. This high rate confirms the strength of the proposed method. In addition, the specificity for COVID-19 class is 100%, meaning that no image from pulmonary and healthy classes has been predicted as COVID-19. We can also remark that sensitivity scores are nearly optimal for the three classes.

To provide further clarifications, we plot the confusion matrix as shown in Fig. 10. As can be seen from Fig. 10, only five (5) COVID-19 images were misclassified, 4 of which are classified as healthy and one as pulmonary. In addition, we can see that there is a slight confusion between images from pulmonary and healthy classes. For instance, 7 images from the pulmonary class were misclassified as healthy, and 5 images from the healthy class were misclassified as pulmonary.

4.4. Studying the tolerance to missing parts of medical image

This aim experiment aims to emulate the scenario of COVID-19 diagnosis via teleradiology. In such a case, radiologists can make remote interpretations of medical images. This will aid in saving time, and human lives and breaking the chain of infection. To achieve an accurate diagnosis, a medical image should be transmitted correctly because transmission errors (e.g., missing parts) can negatively influence the decisions reached by the radiologists. Hence, we evaluate the tolerance of DLNet when there are missing parts in the medical image.

We synthetically mimic this scenario by successively cropping distinct regions in the image. Specifically, we consider ten (10) settings, in each of which a specific region is cropped. In the first five settings, we respectively delete a 100x100 sub-image from the top left, top right, center, bottom right and bottom left regions. In the other settings, we randomly cropped five other sub-images with the same size. Fig. 11 depicts the ten settings. Fig. 12 reports the recognition scores yielded by the different settings.

From Fig. 12, it evidently appears the robustness of our DLNet when there are missing parts in the medical image. Therefore, with such a tolerance, there is no need to retake another clearer image once this scenario occurs. For both random and selected settings, the difference in the scores obtained by using the entire original image was not significant. For instance, by cropping the top left region from the image, the accuracy, specificity and sensitivity have been decreased by 1.86%, 1.24% and 3.59%, respectively. This is may be attributed to the block-wise manner the DLNet uses to learn local features instead of using the entire image. Such a manner alleviates the effect of missed parts by preventing pooling all features in a compacted vector where discriminative features may be dominated by the common global features. In other words, generating the feature vector by considering different image regions allows recompensing the information lost by eliminating certain regions (i.e., the missed regions). Along with the robustness of the proposed DLNet, this experiment demonstrates that different image regions affect the classification decision. However, we can note that

Fig. 13. Processing time, in terms of the number of the filters, required to extract features from one image using the proposed network.

Table 3
Processing time required by each method (in seconds) to extract features for one image.

Method	GoogleNet	VGG-19	VGG-16	ResNet50	ResNet101	DLNet
Processing Time	0.3084	0.5111	0.4267	0.3582	0.6518	0.0410

Table 4
Comparison against the state of the art methods.

Study	Accuracy	Classes	Image type	Number of images
[21]	94.70	COVID / NON-COVID	CT	470
[10]	83.89	COVID/Healthy/Pulmonary	CT	4173
[11]	92.66	COVID-19 severity levels	CXR	909
[16]	95.91	COVID/Healthy/ Streptococcus	CXR	264
[12]	88.70	COVID/Healthy/ Pneumonia	CXR	764
[13]	94.44	COVID/Normal/Bacterial and viral pneumonia	CT/CXR	1200
[3]	97	COVID/Healthy/Pulmonary	CXR	6523
[20]	90.7	COVID/Healthy/ Pneumonia	CXR	1591
[34]	95.16	COVID / NON-COVID	CT	2482
[35]	91.62	COVID / NON-COVID	CXR	1006
[36]	94.7	COVID / NON-COVID	CXR	380
[37]	92	COVID/Healthy/ Pneumonia	CXR	1750
Ours	98.86	COVID/Healthy/Pulmonary	CXR	4000

cropping the top left region has the least effect on the classification results compared to the other settings. This means that its contribution to the classification decision is less than the contributions of the other regions.

4.5. Measuring the processing time

Time is a crucial factor in fighting COVID-19. Quickly identifying infected persons can significantly help prevent other persons, save their lives and break the chain of infection. In view of the high number of infections everyday, the RT-PCR technique for COVID-19 screening seems to be time-consuming, thus, rapid testing methods are highly recommended. The aim of this experiment is to measure the processing time required by the proposed network. Fig. 13 presents the processing time, in terms of the number of filters, required to extract features from one image using our DNLNet.

From the above figure, we can see that processing time is proportional to the number of filters used by the convolution layer of the network. This is because considering more filters will incur additional processing and rises the feature dimensions. For instance, by using 3 filters only, the processing time was 0.022 s, whereas, using 11 filters increases the running time by twice. From these results, we can conclude that DLNet is a real-time network which is suitable for COVID-19 screening, especially in the case of the huge proliferation of this disease.

For a comprehensive analysis, we compare the processing time required by the proposed DLNet and several deep features, as most relevant works have considered using deep architectures for feature learning. In particular, we consider comparing the proposed method with features learned using pre-trained deep networks including GoogleNet, VGG-19, VGG-16, ResNet-50 and ResNet-101 (Table 3).

As can be seen from Table 3, the processing time required by the proposed method is much less than the time required by different deep pre-trained models. For instance, VGG-16 takes 0.4267 s to extract features from one image, which represents the processing time taken by our DLNet multiplied by 10. Similarly, if we look at the other deep networks, we can notice the great deal of time they require compared to DLNet. Deep networks are composed of several stacked layers with a huge number of parameters, which significantly increases the time needed to learn image features. The above-reported results confirm the efficiency of the proposed method, and makes it possible to embed it in frameworks with low computational resources.

4.6. Comparison with related works

We report details of some relevant studies i.e., performance, datasets, type of input image and targeted classes (Table 4). From Table 4, we can note that the targeted classes are not the same for all the cited studies. For instance, the work in [21] has considered two classes namely COVID and non-COVID, while, in [3] three different classes which are COVID, healthy and pulmonary were considered. We can see also that each study used a distinct dataset with a different number of images. In most studies, classification accuracy has exceeded 90%, which suggests that such approaches can be a good alternative to conventional COVID-19 detection methods. It is worth noting that most cited studies are based on deep learning and require a relatively high computational budget. Nevertheless, our proposed method is computationally fast as revealed by the previous experiment. This along with the high classification accuracy it scores (98.86%) makes it a promising solution to fight COVID-19.

5. Conclusion

In this paper, we proposed a fast lightweight network, which is termed as DLNet, for COVID-19 and pulmonary diseases recognition from CXR images. The first layer of DLNet is the convolution layer which acts as a feature detector and uses DCT filters to convolve input images. The main idea behind DLNet is to simultaneously consider filter responses and local binary patterns associated with pixels of the CXR image. DLNet is extracted in a block-wise manner to take advantage of spatial relationships, where the extracted histograms are normalized to cope with illumination changes. We carried out comprehensive experiments on a public dataset. The obtained results revealed the effectiveness of the proposed method as well as the low computational cost it requires. It has also been shown the high tolerance of DLNet to the existence of missed parts in the medical images, makes it possible to integrate DLNet in a telemedicine environment. As a future track, one can design a framework to fuse our proposed lightweight network (e.g., DLNet) with other networks to strengthen individual decisions achieved by each network.

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] https://covid19.who.int/.
[2] F.M. Shah, et al., A comprehensive survey of covid-19 detection using medical images, SN Computer Science 2 (6) (2021) 1–22.
[3] L. Brunese, et al., Explainable deep learning for pulmonary disease and coronavirus COVID-19 detection from X-rays, Comput. Methods Programs Biomed. 196 (2020), 105608.
[4] A. Kumar, et al., SARS-Net: COVID-19 detection from chest x-rays by combining graph convolutional network and convolutional neural network, Pattern Recogn. 122 (2022), 108255.
[5] F. Mohanty, C. Dora, An optimized KELM approach for the diagnosis of COVID-19 from 2D-SSA reconstructed CXR Images, Optik 244 (2021), 167572.
[6] G. Pezzano, et al., CoLe-CNN+: Context learning-Convolutional neural network for COVID-19-Ground-Glass-Opacities detection and segmentation, Comput. Biol. Med. 136 (2021), 104689.
[7] D. Selvaraj, et al., An integrated feature framework for automated segmentation of COVID-19 infection from lung CT images, Int. J. Imaging Syst. Technol. 31 (1) (2021) 28–46.

[8] B. Zheng, et al., MSD-Net: Multi-scale discriminative network for COVID-19 lung infection segmentation on CT, IEEE Access 8 (2020) 185786–185795.

[9] M.K. Hasan, et al., COVID-19 identification from volumetric chest CT scans using a progressively resized 3D-CNN incorporating segmentation, augmentation, and class-rebalancing, Inf. Med. Unlocked 26 (2021), 100709.

[10] Alshazly, H., et al., *COVID-Nets: Deep CNN Architectures for Detecting COVID-19 Using Chest CT Scans.* medRxiv, 2021.

[11] H. Aboutalebi, et al., Covid-net cxr-s: Deep convolutional neural network for severity assessment of covid-19 cases from chest x-ray images, Diagnostics 12 (1) (2022) 25.

[12] K. Upadhyay, M. Agrawal, D. Deepak, Ensemble learning-based COVID-19 detection by feature boosting in chest X-ray images, IET Image Proc. 14 (16) (2020) 4059–4066.

[13] A.S. Elkorany, Z.F. Elsharkawy, COVIDetection-Net: A tailored COVID-19 detection from chest radiography images using deep learning, Optik 231 (2021), 166405.

[14] M.M. Rahman, et al., HOG+ CNN Net: Diagnosing COVID-19 and pneumonia by deep neural network from chest X-Ray images, Sn Computer Science 2 (5) (2021) 1–15.

[15] V. de Carvalho Brito, et al., COVID-index: A texture-based approach to classifying lung lesions based on CT images, Pattern Recogn. 119 (2021), 108083.

[16] K. Shankar, et al., Deep learning and evolutionary intelligence with fusion-based feature extraction for detection of COVID-19 from chest X-ray images, Multimedia Syst. (2021) 1–13.

[17] S.-H. Wang, X. Zhang, Y.-D. Zhang, DSSAE: Deep stacked sparse autoencoder analytical model for COVID-19 diagnosis by fractional Fourier entropy, ACM Transactions on Management Information System (TMIS) 13 (1) (2021) 1–20.

[18] W. Shui-Hua, et al., Deep rank-based average pooling network for COVID-19 recognition, Computers, Materials, & Continua (2022) 2797–2813.

[19] Y.-D. Zhang, et al., MIDCAN: A multiple input deep convolutional attention network for Covid-19 diagnosis based on chest CT and chest X-ray, Pattern Recogn. Lett. 150 (2021) 8–16.

[20] Osman Ilhan, H., G. Serbes, and N. Aydin, *Decision and Feature Level Fusion of Deep Features Extracted from Public COVID-19 Data-sets.* arXiv e-prints, 2020: p. arXiv: 2011.08528.

[21] D. Li, Z. Fu, J. Xu, Stacked-autoencoder-based model for COVID-19 diagnosis on CT images, Applied Intelligence 51 (5) (2021) 2805–2817.

[22] X. Zhao, et al., D2A U-Net: Automatic segmentation of COVID-19 CT slices based on dual attention and hybrid dilated convolution, Comput. Biol. Med. 135 (2021), 104526.

[23] D. Müller, I. Soto-Rey, F. Kramer, Robust chest CT image segmentation of COVID-19 lung infection based on limited data, Inf. Med. Unlocked 25 (2021), 100681.

[24] O. Elharrouss, N. Subramanian, S. Al-Maadeed, An encoder–decoder-based method for segmentation of COVID-19 lung infection in CT images, SN Computer Science 3 (1) (2022) 1–12.

[25] H. Chen, et al., Unsupervised domain adaptation based COVID-19 CT infection segmentation network, Applied Intelligence (2021) 1–14.

[26] J. Zhang, et al., Dense GAN and multi-layer attention based lesion segmentation method for COVID-19 CT images, Biomed. Signal Process. Control 69 (2021), 102901.

[27] C.J. Ng A.B.J. Teoh DCTNet: A simple learning-free approach for face recognition 2015 IEEE.

[28] Y. Zhang, et al., ICANet: a simple cascade linear convolution network for face recognition, EURASIP Journal on Image and Video Processing 2018 (1) (2018) 1–7.

[29] A. Korichi, S. Slatnia, O. Aiadi, TR-ICANet: A Fast Unsupervised Deep-Learning-Based Scheme for Unconstrained Ear Recognition, Arabian Journal for Science and Engineering (2022) 1–12.

[30] X. Wang, et al., Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.

[31] M.E. Chowdhury, et al., Can AI help in screening viral and COVID-19 pneumonia? IEEE Access 8 (2020) 132665–132676.

[32] https://www.kaggle.com/nih-chest-xrays/sample/version/4.

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

[34] X. Ma, et al., COVID-19 lesion discrimination and localization network based on multi-receptive field attention module on CT images, Optik 241 (2021), 167100.

[35] A.K. Das, et al., Automatic COVID-19 detection from X-ray images using ensemble learning with convolutional neural network, Pattern Anal. Appl. (2021) 1–14.

[36] A.M. Ismael, A. Şengür, Deep learning approaches for COVID-19 detection based on chest X-ray images, Expert Syst. Appl. 164 (2021), 114054.

[37] T. Ozturk, et al., Automated detection of COVID-19 cases using deep neural networks with X-ray images, Comput. Biol. Med. 121 (2020), 103792.