EURASIP Journal on Advances
in Signal Processing

RESEARCH

Open Access

Performance evaluation of lung sounds classification using deep learning under variable parameters

Zhaoping Wang[1*] and Zhiqiang Sun[2]

*Correspondence:
wzp2008@189.cn

[1] Department of Emergency, Qilu Hospital of Shandong University (Qingdao), Qingdao, China
[2] School of Automation, Qingdao University, Qingdao, China

Abstract

It is desired to apply deep learning models (DLMs) to assist physicians in distinguishing abnormal/normal lung sounds as quickly as possible. The performance of DLMs depends on feature-related and model-related parameters heavily. In this paper, the relationship between performance and feature-related parameters of a DLM, i.e., convolutional neural network (CNN) is analyzed through experiments. ICBHI 2017 is selected as the lung sounds dataset. The sensitivity analysis of classification performance of the DLM on three parameters, i.e., the length of lung sounds frame, overlap percentage (OP) of successive frames and feature type, is performed. An augmented and balanced dataset is acquired by the way of white noise addition, time stretching and pitch shifting. The spectrogram and mel frequency cepstrum coefficients of lung sounds are used as features to the CNN, respectively. The results of training and test show that there exists significant difference on performance among various parameter combinations. The parameter OP is performance sensitive. The higher OP, the better performance. It is concluded that for fixed sampling frequency 8 kHz, frame size 128, OP 75% and spectrogram feature is optimum under which the performance is relatively better and no extra computation or storage resources are required.

Keywords: Lung sounds classification, Convolutional neural network, Spectrogram, Mel frequency cepstrum coefficients

1 Introduction

At present, it is reported by World Health Organization (WHO) that lung or respiratory disease is still one of the most ordinary causes of death worldwide, especially in low income countries [1]. And it is estimated that more than 3 million people died of various respiratory diseases each year [2]. Lung or breath sounds may change from normal to abnormal state because of lung or respiratory disease(s). Abnormal lung conditions can be screened out and intervened timely with the help of lung sounds auscultation. However, the diagnostic accuracy depends on the physician's knowledge and experience heavily. An experienced physician may distinguish abnormal lung sounds from normal ones quickly and make further examination plans accurately. However, this is not the case for inexperienced physicians. If computers can be used to diagnose abnormal lung sounds, it would be very helpful. With the development of computation and electronic technology, it is coming true. First of all, respiratory sounds can be recorded by electronic stethoscopes, and be stored as audio files for further processing. We can collect lots of normal and abnormal lung sounds by the electronic instrument. After accumulating sufficient data, we can try to develop a model to classify normal/abnormal lung sounds, or even to diagnose lung diseases automatically.

Thanks to the development of machine learning, especially deep learning, computer-aided lung sounds detection technology has made rapid progress. There have been many works about detecting abnormal lung sounds via machine learning or deep learning. Until 2015, the machine learning methods such as support vector machine (SVM), principal component analysis (PCA) had played major roles [3–8]. And later after this year, the deep learning model, especially convolutional neural network (CNN), had been introduced to this field and showed to be superior to machine learning on accuracy and generalization ability [9–14]. For machine learning, the so-called hand-crafted features, namely the peculiar signatures of some abnormal/normal lung sounds, must be extracted as the input to the learning model in advance. Various features, such as skewness and kurtosis of time signal or spectral density in frequency domain are extracted from sounds [8]. Machine learning does not require a large number of samples, but has one big drawback of the limited generalization ability.

Deep learning is an end-to-end approach and does not need the step of features extraction. The raw samples are fed to deep learning models (DLMs) directly. In recent years, it has been applied to speech recognition, object recognition, classification and other fields successfully [14]. In the field of biomedicine, Alpha Fold of Deep Mind can accurately predict the structure of the human proteome (a collection of all proteins encoded by the human genome), and the resulting dataset covers the structural positions of nearly 60% of the amino acids in the human proteome prediction, and the prediction results have a high degree of confidence [15]. On the other hand, deep learning is widely used in the field of diagnosis. As concluded by Fourcade et al. [16], DLMs "will contribute to optimize routine tasks and thus have a potential positive impact on our practice."

Through an investigation in the corresponding author's hospital, we found that an electronic stethoscope able to make initial classification of lung sounds would be very welcome by physicians. It could get rid of some troubles of the traditional in-ear stethoscope by transferring the sounds to a computer, or even a mobile phone. The ability of lung sounds classification can lighten the burden of physicians to a great extent. An issue concerned universally by physicians is the accuracy and practicality.

Over the past 2 decades, there have been many works about lung sounds classification by using machine learning or deep learning. Lots of solutions with different parameters and performance levels were presented. The parameters were selected and set empirically in many works. There are few works discussing how the performance is affected by different parameters. In this paper, we will focus on this topic and try to discover the relationship between the parameters and performance.

The remainder of this paper is organized as follows. A detailed literature review is performed, in which some representative works on automatic classification of lung sounds are reviewed. Then ICBHI 2017 dataset is introduced briefly, and the emphasis is put on data preprocessing and augmentation. The CNN model is discussed from the aspects of architecture, features and parameters in detail. A comparative study of classification performance in the proposed work versus up-to-date ones is performed. The summary of the paper and outlook of future work is presented in the last section.

A CNN model includes many parameters, such as feature-related and model-related ones. Before training it, we have to select appropriate parameters for it. On one hand, the parameters can be designed by trial and error approach. On the other hand, we can inherit some existing parameters proved to be effective by other works. It would be very helpful for parameter selection if we clarify the relationship between the performance and parameters of the CNN. It is the aim of this work and has attracted few researchers' attention. The length of lung sounds frame, overlap percentage (OP) of successive frames and feature types are picked as three typical parameters. And the relationship between these parameters and classification performance is explored in detail through experiments. This is the main contribution of this study.

It must be pointed out that the CNN model used in this work had been validated by other work [17], and so the problem of tuning hyperparameters of the network such as number of filters and layers, activation function selection is out of scope of this work.

2 Literature review

There have been lots of works about automatic lung sounds classification via DLMs. Several respiratory sound datasets have been used to train and test DLMs. The signals were collected from patients and healthy volunteers by using an electronic stethoscope or microphone. Some datasets are publicly available, while others are limited to personal use. To the best of the authors' knowledge, we have collected some frequently used datasets shown in Table 1.

Among these datasets, RespiratoryDatabase@TR and ICBHI 2017 are two of the most popular ones. The former was created by Altan et al. [20], including not only sound audio signals but the chest X-ray films and pulmonary function test (PFT) measurements of related subject. RespiratoryDatabase@TR has been widely used to assess the severity of chronic obstructive pulmonary disease (COPD) [26, 27]. The later was originally compiled to support the scientific challenge organized at Int. Conf. on Biomedical Health

Table 1 Some frequently used lung sounds datasets

No.	Dataset	Year	Number of records/subjects	Publicly available	Creator
1	ICBHI 2017	2017	920/126	Yes	Rocha et al. [17]
2	SPRSound	2022	2683/292	Yes	Zhang et al. [18]
3	HF_Lung_V2	2022	13964/300	Yes	Hsu et al. [19]
4	Respiratory Database@TR	2021	3696/77	Yes	Altan et al. [20]
5	R.A.L.E Lung Sounds 3.1	2004	$\geq$ 50/unknown	Commercial use	PixSoft and Medi-Wave [21]
6	Own generated	2023	1371/1371	No	Aptekarev et al. [22]
7	Own generated	2021	1918/871	No	Kim et al. [10]
8	Own generated	2022	301/103	No	Fraiwan et al. [23]
9	Own generated	2022	287/47	No	Tessema et al. [24]
10	Own generated	2020	17930/1630	No	Aykanat et al. [11]
11	Own generated	2023	1021/126	No	Choi et al. [25]

Informatics and is freely available to everyone [17]. It has been used by many works to train and test DLMs and will be utilized in this work.

Based on these datasets, researchers have tried to distinguish between normal and abnormal lung sounds automatically via machine learning or deep learning. In recent years, deep learning models have been playing the major role. For comparison, we sort some works about lung sounds classification according to the datasets and classification models used, as shown in Table 2.

From Table 2, it can be seen that spectrogram-like features were used more widely, including but not limited to spectrogram, mel-spectrogram, log-spectrogram, scalogram. Some works fused spectrogram and mel frequency cepstrum coefficients (MFCCs) as features for DLMs with the intention of improving classification performance. In addition, the quantity of deep learning-related works is far more than machine learning-related ones.

When ICBHI 2017 was used as dataset to train and test a DLM, Acharya et al. [28] achieved a score of 71.81% on four-class classification by re-training a deep CNN-RNN (recurrent neural network) model with patient specific data. Chen et al. [29] trained a deep residual network (ResNet) for triple classification of respiratory sounds with the accuracy, sensitivity, and specificity up to 98.79%, 96.27% and 100%, respectively. It was reported that the proposed model outperformed CNN. Shuvo et al. [30] used empirical mode decomposition (EMD) and the continuous wavelet transform (CWT) to train a lightweight CNN with the accuracy scores of 98.92% for three-class chronical classification and 98.70% for six-class pathological classification, respectively, which outperformed some larger network and other contemporary lightweight models. Petmezas et al. [12] made a four-class lung sounds classification using a hybrid CNN-LSTM (long short-term memory) network and spectrogram as feature. They achieved the scores as high as sensitivity 52.78%, specificity 84.26%, score 68.52% and accuracy 76.39%. Cinyo et al. [31] combined a CNN architecture with support vector machine (SVM)/softmax as an architecture to which various classifiers were incorporated. It was reported that the best classification accuracy was 83% with VGG16-CNN-SVM model. Jayalakshmy et al. [32] employed conditional generative adversarial networks and made four-class classification using a pre-trained CNN (ResNet-50) and scalogram as feature, achieving an accuracy of 92.5%. Asatani et al. [33] used an improved convolutional RNN as a quadruple classifier of lung sounds and yielded the results of sensitivity 0.63, specificity 0.83 and

Table 2 Summary of studies conducted on lung sounds classification

Dataset	Classification model	Features	Refs.
ICBHI 2017	Deep learning	Spectrogram mel-spectrogram scalogram, etc.	[12, 28–33]
		MFCCs	[34–36]
		Both the above	[37, 38]
	Machine learning	Spectrogram mel-spectrogram scalogram, etc.	[39, 40]
		MFCCs	[41]
Respiratory Database@TR	Deep learning	Spectrogram	[27, 42]
		Time-domain features	[43, 44]
	Machine learning	Empirical wavelet transform	[45]

score 0.72. Some works used MFCCs as feature to train/test the deep learning network. Perna [34] used a deep CNN architecture with regularization to classify the breathing cycles into three classes: healthy, chronic and non-chronic, and obtained the scores of accuracy 82%, precision 87%, recall 83% and F1_score 84%. Dhavala et al. [35] made triple classification of lung sounds by employing a CNN and achieved test accuracy 86.25%. Saraiva et al. [36] proposed a CNN to make a quadruple classification and obtained accuracy 74.3%. A feature-based fusion network with three features, i.e., spectrogram, MFCCs, and chromagram was proposed to classify lung sounds as six categories by Tariq et al. [37], leading to the highest accuracy 99.1%. Rishabh et al. [38] obtained the statistical features from MFCCs, mel-spetrogram, chroma STFT, etc. of lung sounds, and fed them to a CNN and got a quadruple-classification model with accuracy 75.04%. Tasar et al. [39] proposed a mixed model to generate features and applied decision tree (DT), SVM, and k nearest neighbors (KNN) to classify lung sounds. The results showed that KNN outperformed the other two classifiers. Ari et al. [40] used SVM as a quadruple classifier lung sounds with the scalogram as feature, and achieved accuracy 72.69%. Jakovljević et al. [41] used MFCCs as feature and hidden Markov model with Gaussian mixture model as the quadruple classifier of lung sounds, and the best score 39.56% was reported.

In most cases, RespiratoryDatabase@TR was used to detect the severity of Chronic Obstructive Pulmonary Diseases (COPD). Roy et al. [27] generated mel-spectrogram snippet representation as input feature and compared the performance of two classifiers for COPD severity detection. Yu et al. [42] extracted bispectrum of lung sounds as feature of the CNN classifier, to assist diagnosis of COPD. Altan et al. [43] applied the deep belief networks (DBN) to separate the lung sounds from different levels of COPD with extracting 3D second order difference plot in time domain as feature. In their another work [44], the statistical features of frequency modulations were extracted using Hilbert–Huang transform and then were fed to a DLM. Ahmet et al. [45] extracted statistical features using empirical wavelet transform (EWT) algorithm and then applied them to SVM, AdaBoost, random forest and J48 DT, respectively, in aid of diagnosis of COPD.

In some works, lung sounds were collected by physicians in field to generate private datasets, to see Table 1. It is inappropriate to compare the performance of these works because of these varied datasets, and they will not be reviewed in detail.

3 Materials and methods

3.1 Data preparation and preprocessing

The ICBHI 2017 database consists of about 5.5 h of data recording sampled from 126 subjects totally, which contains 6898 respiratory cycles (i.e., from inspiratory to expiatory phase), from which 3642 contain normal sounds, 1864 contain crackles, 886 contain wheezes, and 506 contain both crackles and wheezes can be extracted as shown in Table 3.

The audio samples in the dataset were sampled at frequencies of 4 kHz, 10 kHz and 44.1 kHz using different instruments. There are differences in amplitudes of audio signals across instruments. Another challenge is that noise exists, such as rubbing sound of the stethoscope with the participant’s dress and ambient noise. In addition, lung sounds are always presented along with heartbeat sounds.

Table 3 Distribution of classes in ICBHI 2017

Type	Amount	Ratio (%)
Normal	3642	52.80
Crackle	1864	27.02
Wheeze	886	12.84
Crackle + Wheeze	506	7.33
total	6899	100

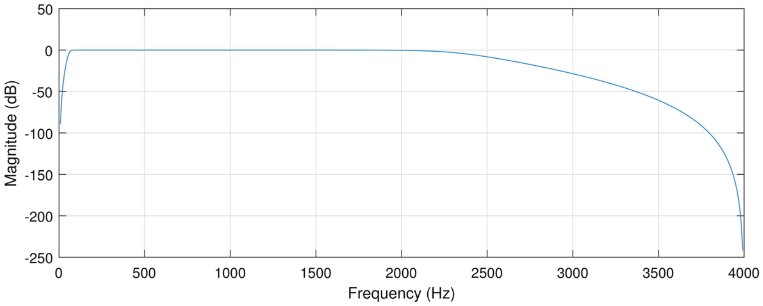


Fig. 1 Magnitude characteristic of the band-pass filter

The length of respiratory cycles in the dataset varies in a wide range from 0.3 to 12 s. In theory, a respiratory cycle takes 3–5 s. Such varieties in the data make it challenging to classify the lung sounds. Therefore, the raw signals in the dataset must be preprocessed before being brought to train/test a DLM.

3.1.1 Resampling the signals

First of all, the audio signals are resampled at a frequency of 8 kHz to standardize the signal length. It is noted that the maximum frequency of lung sounds is not greater than 3 kHz [46], so the resampling operation could not lead to important information loss of the audio sounds.

3.1.2 Noise filtering

In order to mitigate the effect of ambient noise and heartbeat sounds, the lung sounds samples are filtered by a 10th Butterworth band-pass filter [38]. The transfer function $H(z)$ of the filter is

$$H(z) = \prod_{i=1}^{5} H_i(z);\ H_1(z) = \frac{0.6749 + 0.6749z^{-2}}{1 - 1.9700z^{-1} + 0.9722z^{-2}};$$
$$H_2(z) = \frac{0.6749 + 0.6749z^{-2}}{1 + 0.3569z^{-1} + 0.5496z^{-2}};\ H_3(z) = \frac{0.5752 + 0.5752z^{-2}}{1 - 1.9236z^{-1} + 0.9258z^{-2}}$$
$$H_4(z) = \frac{0.5752 + 0.5752z^{-2}}{1 + 0.2364z^{-1} + 0.1236z^{-2}};\ H_5(z) = \frac{0.5473 + 0.5473z^{-2}}{1 - 0.8529z^{-1} - 0.0945z^{-2}}$$

The magnitude characteristic is shown in Fig. 1. It can be seen that the bandwidth of the filtered samples is about [25, 2500] Hz. For the frequency of heartbeat is far below 20 Hz, the heart sounds can be filtered out completely. In general, the frequency of background or electronic noise is concentrated below 50 Hz, so they can be also eliminated by this filter. Some ambient noise above 2500 Hz can also be filtered out.

3.1.3 Normalization

After the noises being filtered out, all signals are normalized to the range $[-1, 1]$ for standardizing the data across different recording devices.

3.1.4 Data segmentation

In accordance with the annotated respiratory cycle, each audio signal is segmented timing with a 5 s duration. If the time duration of one annotated respiratory cycle is not over 5 s, the length of related audio clip will be extended to 5 s by sample padding. According to Fraiwan [23], it is appropriate to select 5 s as the cycle time, for it can cover both faster and slower breathing rates without adding extra complexity of the model.

3.1.5 Transformation of time series to spectrogram-like feature

It is not recommended to apply the lung sound samples to CNNs as feature directly. On one hand, there may be a significant difference in the waveforms of two time series with the same label, especially when the two series are disturbed by noise. On the other hand, the major disadvantage of CNNs on time series is the use of Euclidean kernels. The kernel considers only a continuous and short time series subsequence at a time. In order to extract more representative features, non-contiguous and longer time series samples must be analyzed. To overcome these drawbacks, the time series of lung sounds are transformed to spectrogram-like images as features to CNNs.

A spectrogram is a two-dimensional image that shows the change of sound amplitude with frequency and time as shown in Fig. 2.

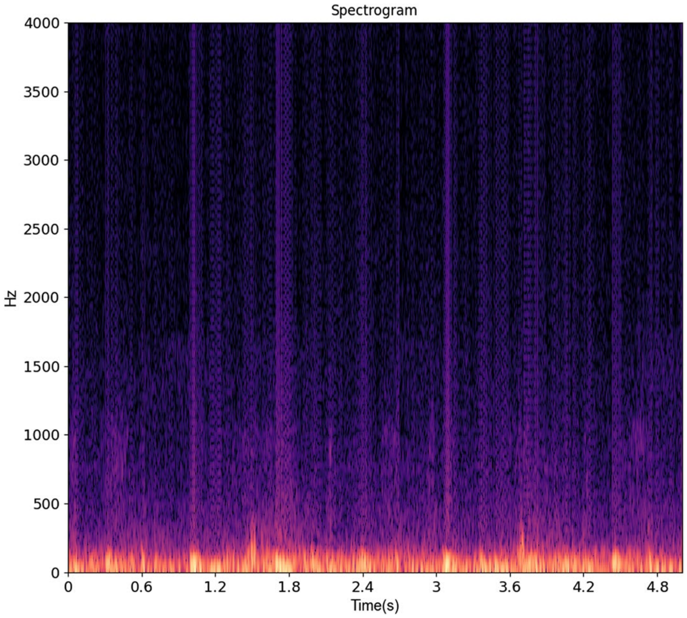


Fig. 2 An example of lung sound spectrogram

The vertical and horizontal axes represent frequency and time, respectively. Not only does the spectrogram match our understanding of sounds through frequency decomposition, but it also allows us to use 2-dimensional analysis architectures. The fact that spectrogram certainly is the best-suited representation of audio signals for analysis has become the common view.

3.2 Data augmentation

As mentioned in Table 3, we have extracted 1864 cycles containing crackles, 886 containing wheezes, 506 containing both crackles and wheezes, and 3643 containing normal sounds. Obviously, the number of different types of records is unbalanced, which may lead to overfitting during model training and poor generalization ability. So, the unbalance must be corrected to keep the number of four types of records even. One straightforward solution is to delete some records from the type with greater number of records randomly, in order to make the number approach the type with less records. This will waste a lot of useful data. Another effective solution is to expand the capacity of data, which is very common in image processing, for example, to increase the sample capacity through image dithering, inversion, rotation, etc.

There are some popular ways for audio data augmentation, such as time stretching, pitch shifting and background noise inserting [47]. By time stretching, we can slow down or speed up the audio samples while keeping the pitch unchanged. On other hand, the pitch of the audio samples can be raised or lowered while keeping the duration unchanged. By mixing the audio samples with some background noise signal, we can get a new record of sample augmented [48]. We will use these three approaches to augment the lung sounds for balance. First of all, the white noise is selected and inserted to the lung sounds. White noise consists of some random sound samples with similar amplitude but various frequencies. The performance of speech emotion recognition can be increased by addition white noise to original sound [49]. Lung sounds are very weak, and the signal-to-noise ratios (SNRs) of many records are not high. So, the SNR should be controlled not to be too low when noise is inserted. Three SNRs of 10 dB, 15 dB and 20 dB are determined by trail and error.

In order to achieve balance in the number of records labeled by the four labels, the number of records labeled by crackle should be expanded to be twice as the original number, and the number of wheeze records should be expanded to be four times, and then the number of records both with crackle and wheeze should be seven times. For the records with the two latter labels, the augmentation is a little exaggerated. To avoid it, we formulate a procedure for augmentation as follows:

(1) Every time to train the CNN model, first of all, we select 400 from 506 records with both crackle and wheeze randomly and then to add noise to the 400 records under the SNRs of 10 dB, 15 dB and 20 dB, respectively. So we get 1200 "new" records meaning that the number of records with both crackle and wheeze is expanded to 1700 approximately.

(2) Every time to train the CNN model, first of all, we select 750 from all the records with wheeze randomly and then to add noise to the 750 records under the SNRs of

Table 4 Distribution of classes in ICBHI 2017 after white noise addition

Type	Amount	Ratio (%)
Normal	8000	27.08
Crackle	8176	27.67
Wheeze	6544	22.15
Crackle + Wheeze	6824	23.10
Total	295,44	100

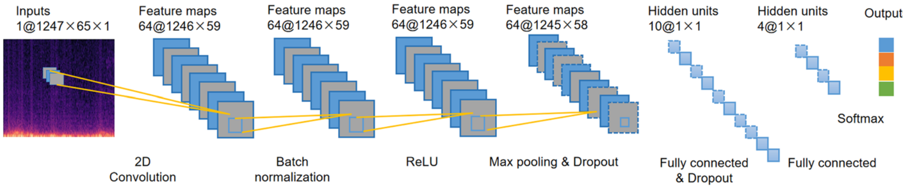


Fig. 3 The architecture of the CNN with one certain parameters setting

10 dB, 15 dB and 20 dB, respectively. It means that the number of records labeled by wheeze is expanded to about 1600.

(3) Every time to train the CNN model, we select 180 records from crackle records randomly and divide them into three parts evenly, and then each part has 60 records. The three parts are disturbed by noised under three SNRs of 10 dB, 15 dB and 20 dB, respectively. It means that the number of crackle records is expanded to about 2000.

(4) Every time to train the CNN model, 2000 records are selected from normal record randomly as training and test data. Finally, the distribution of augmented lung data is shown in Table 4. It can be seen that the unbalanced number of the four types of records has been corrected.

For the balanced dataset, time stretching and pitch shifting are performed successively. Each record in the balanced dataset is stretched by two factors: $\{0.93, 1.07\}$ and pitch shifted by four values: $\{-1, 1\}$. Finally, we have four times as many number of each types of lung sounds in the augmented dataset as in Table 4.

3.3 Architecture of deep learning model

A relatively simple CNN is introduced, including input layer, pooling layer, batch normalization layer, max pooling layer and fully connected layers in sequence. There are two fully connected layers. Two dropout layers are inserted before and after the first one, respectively. The second fully connected layer is followed by the second dropout layer, and connected to output layer. This architecture shown in Fig. 3 has been validated by Rocha et al. [50]. The softmax function is adopted in the output layer, as shown in Fig. 3.

The hyperparameters of the CNN to be used are listed in Table 5. In order to ensure the consistency of the CNN architecture among different parameter settings, the hyperparameters are kept fixed and not to be tuned during the training stage.

Table 5 The architecture of the CNN

Parameter setting	SpectroGram (SG)		Mel frequency cepstrum coefficients (MFCCs)	
	Activations	**Parameters**	**Activations**	**Parameters**
Input	$m \times n \times 1$	–	$m \times n \times 1$	–
2D Convolution	$K1 \times K2 \times 64$	Convolution size: 7 Stride: 1 Padding: 0 Filters number: 64	$J1 \times J2 \times 64$	Convolution size: 3 Stride: 1 Padding: 0 Filters number: 64
Batch normalization	$K1 \times K2 \times 64$	–	$J1 \times J2 \times 64$	–
ReLU	$K1 \times K2 \times 64$	–	$J1 \times J2 \times 64$	–
Max Pooling	$K3 \times K4 \times 64$	Pooling size: 2 Stride: 1	$J3 \times J4 \times 64$	Pooling size: 2 Stride: 1
Dropout	$K3 \times K4 \times 64$	50%	$J3 \times J4 \times 64$	50%
Fully connected	$1 \times 1 \times 10$	Size: 10	$1 \times 1 \times 10$	Size: 10
Dropout	$1 \times 1 \times 10$	50%	$1 \times 1 \times 10$	50%
Fully connected	$1 \times 1 \times 4$	Size: 4	$1 \times 1 \times 4$	Size: 4
Softmax	$1 \times 1 \times 4$	–	$1 \times 1 \times 4$	–

$K1 = m - 6, K2 = n - 6, K3 = K1 - 1, K4 = K2 - 1, J1 = m - 2, J2 = n - 2, J3 = J1 - 1, J4 = J2 - 1$

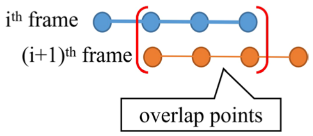


Fig. 4 The overlap points between successive frames

3.4 Generation of features

Two types of features, spectrogram and MFCCs, are consider in this paper. We cannot get them without fast Fourier transform (FFT). First of all, the sound samples of a respiratory cycle must be windowed to some successive frames on which FFTs are performed. The window length (Lwin) of FFT should not exceed 40 ms in general because of the nonstationarity of lung sound. In discrete domain, if the sampling frequency is fixed to 8 kHz, the window size or the frame length should not be above 320 points. There should be some overlap between successive frames in order to keep continuity. The overlap percentage (OP) is the ratio between the number of overlap points and frame length. For example, as in Fig. 4, the frame length is 4, and the overlap length 3, and then the overlap percentage is $3/4 = 75\%$.

The number of frames of a respiratory cycle lasting for 5 s can be calculated as

$$m = \left\lfloor \frac{40000 - \mathrm{Lwin}}{\mathrm{Lwin} \times (1 - \mathrm{OP})} + 1 \right\rfloor \tag{1}$$

when the sampling frequency is $f_s = 8$ kHz.

For each frame $x(n), n = 0, 1, \ldots, \mathrm{Lwin} - 1$, FFT is performed according to

$$X(k) = \sum_{n=0}^{\mathrm{Lwin}-1} x(n) \exp\left(-j\frac{2\pi}{N}nk\right), \quad k = 0, 1, \ldots, \mathrm{Lwin} - 1 \tag{2}$$

where k is the index of frequency bin. The magnitude $|(X(k))|$ is used to construct a spectrogram. Considering the conjugate symmetry of Fourier transform, it is only necessary to take the first half of the magnitudes of FFT. It means that we only keep a n-dimension vector $n = \mathrm{Lwin}/2 + 1$ after performing one FFT. After all the frames are transformed by FFT, a $n \times m$ matrix, then the spectrogram is built, as shown in Fig. 2. The column vectors are corresponding to the FFT magnitudes of the frames of one respiratory cycle, respectively.

Another popular spectrogram-like feature employed in respiratory sounds classification is MFCCs, which is also applied to speech recognition frequently. MFCCs are introduced to separate the speech signal spectrum $S(z)$ into the source $U(z)$ (periodic signal generated by opening and closing of the vocal folds which generates the pitch) and vocal tract filter $H(z)$ which changes according to the word being spoken. The speech spectrum can be represented as

$$S(z) = U(z) \times H(z) \tag{3}$$

This equates to the convolution of the source with the vocal tract filter in time domain:

$$s(n) = u(n) * h(n) \tag{4}$$

where $s[n]$, $u[n]$ and $h[n]$ are the speech, source and filter responses, respectively.

MFCCs incorporate the fact that the human auditory system is more sensitive to changes at lower frequencies (linear below 1000 Hz) than at higher frequencies (logarithmic above 1000 Hz). To model human pitch perception, a series of triangular filter banks are applied to the speech spectrum which are spaced linearly below 1000 Hz and logarithmically above 1000 Hz according to the mel scale which is given as

$$f_{\mathrm{mel}} = 2595 * \log_{10}(1 + f/700) \tag{5}$$

where f_{mel} is the frequency converted in the mel scale and f is frequency in the linear domain.

There are four steps for calculating MFCCs as follows:

Step 1 To window the lung sound signal and perform STFT to the windowed frame, and get its spectrum $Y(k) = X^2(k), k = 0, 1, \ldots, N-1$. N is the frame size or STFT length.

Step 2 Apply mel-scaled filter bank to the spectrum as shown in Fig. 5. In this bank with the same bank height, the number of filters is $M = 20$, and the sampling frequency is 8 kHz.

Step 3 Calculate the log of the summed filter bank energies:

$$m_j = \log\left(\sum_{k=0}^{N-1} Y(k) * H_j(k)\right), \quad 1 \leq j \leq M \tag{6}$$

Step 4 The discrete cosine transform (DCT) of the log values is calculated to give the coefficients as in Eq. (7):

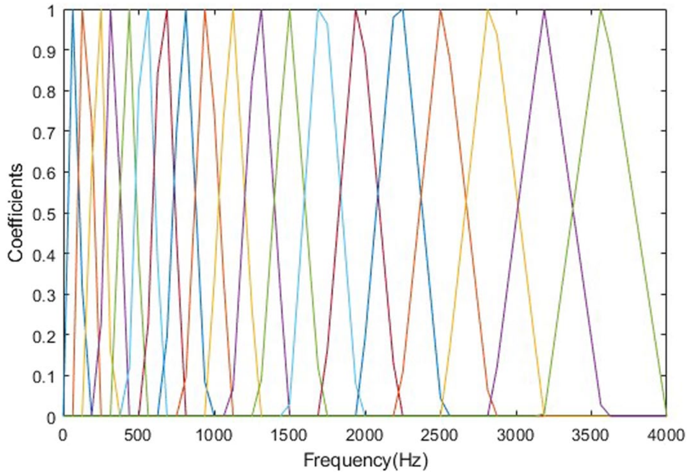


Fig. 5 Mel-scale filter bank

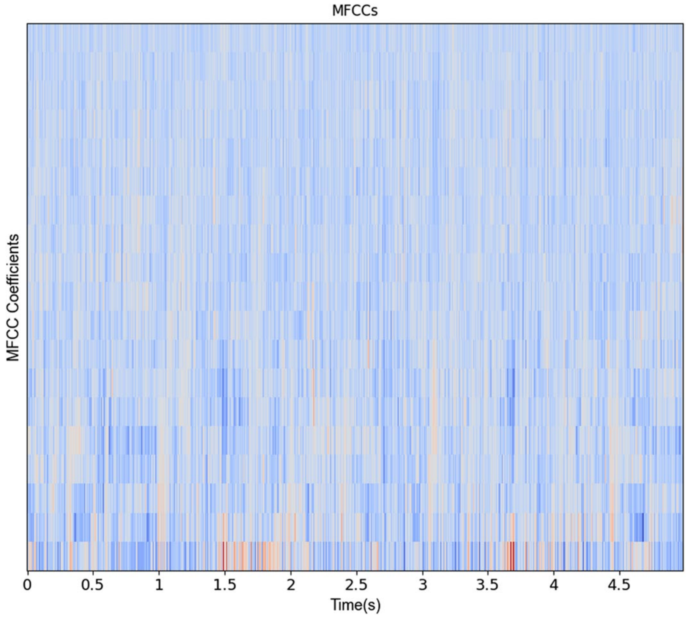


Fig. 6 An example of MFCCs image

$$c_i = \sqrt{\frac{2}{M}} \sum_{j=1}^{M} m_j \cos\left[\frac{i\pi}{M}(j-0.5)\right], \quad i = 1, 2, \ldots, L \tag{7}$$

where L is the order of MFCCs and usually lies in interval [2,15]. We choose 15 for it considering the granularity of MFCCs. M is the number of filters in the bank.

Following the procedures above, we can get an example of MFCCs image for a segment of lung sound, to see Fig. 6. It should be noted that the size of MFCCs image depends on the signal duration and the order of MFCCs.

3.5 Platform and parameters setting

In this paper, the CNN for lung sounds classification is implemented with Pytorch in Python and tested on our workstation with a 64-bit Windows 10 with Intel i7-7800X 3.50 GHz processor and an Nvidia GTX 3060 graphics card.

We attempt to discover the relationship between classification performance and parameters of the CNN. The Lwin, OP, feature type (spectrogram or MFCCs) are selected as comparative parameters, as shown in Table 6.

The parameters in Table 6 may have different value combinations. There are totally 18 different combinations. As mentioned above, the image size of feature $m \times n$ depends on the parameter setting. For example, if Lwin $= 128$, OP $= 75\%$ and feature type is SG, we have $m = 1247$ and $n = 65$. If feature type is MFCCs, m is still equal to 1247, but n is kept to be 15 no matter what the value of Lwin is.

Classification performance is evaluated with tenfold cross-validation. Ninety percent of the data is used for training and 10% for validation to avoid overfitting [26]. It is a common practice. We partition the dataset by patients and not by lung sounds. None of lung sounds from the same patient is used in both training and validation set. The validation set in each fold contains at least one class for every possible recording location.

At training stage, the CNN uses Adam optimizer with learning rate 2×10^{-4} and gradient decay factor of 0.5.

3.6 Performance criteria

The following five typical performance criteria, including accuracy, specificity, sensitivity, precision and F1_score are selected to evaluate the CNN model:

$$\text{Accuracy} = \frac{C_c + C_w + C_b + C_n}{T_c + T_w + T_b + T_n};\ \text{Specificity} = \frac{C_n}{T_n};\ \text{Sensitivity} = \frac{C_c + C_w + C_b}{T_c + T_w + T_b};$$
$$\text{Precision} = \frac{C_c + C_w + C_b}{C_c + C_w + C_b + T_n - C_n};\ \text{F1\_score} = 2 * \frac{\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}}$$

where $C_i, T_i, i = c, w, b, n$ refers to the number of correctly recognized instances of class i, and the total number of instances of class i in the test (or validation) set, respectively. The symbols c, w, b and n stand for the class of crackle (c), wheeze (w), c and w and normal, respectively [41].

Sensitivity (true-positive rate) refers to the probability of a positive test, conditioned on truly being positive. On the other side, specificity (true-negative rate) refers to the probability of a negative test, conditioned on truly being negative. Sensitivity is an indicator of the ability to correctly identify those with disease, and specificity is used to indicate the ability correctly identify those without disease. Accuracy is a measure of how well a binary classification test correctly determines whether a patient is healthy or not.

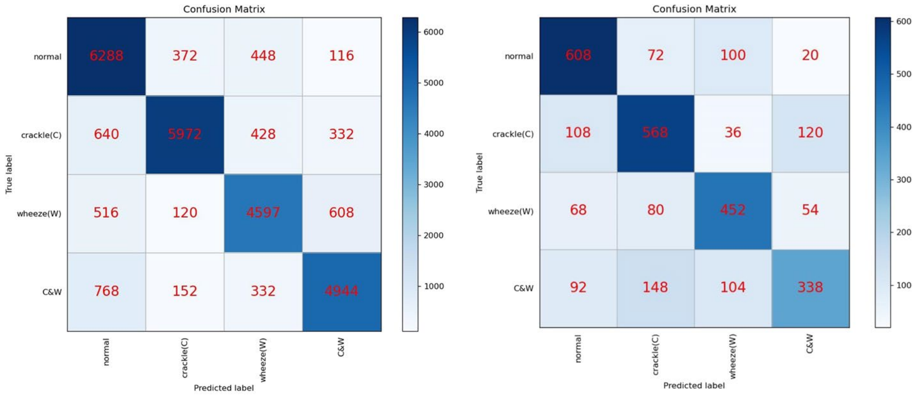
Table 6 Parameter and its possible values

Parameter	Value		
Window length (Lwin)	64	128	256
Overlap percentage (OP)	75%	50%	25%
Feature type	Spectrogram (SG)	MFCCs	

Table 7 Performance criteria with SG feature

Parameter combination	Performance criteria (%)					Stage
	Sensitivity (CI)	Specificity (CI)	Accuracy (CI)	Precision (CI)	F1_score (CI)	
Lwin = 64 OP = 25%	62.3 (60.1–65.1)	64.5 (61.8–66.7)	62.9 (60.6–65.4)	82.4 (79.1–89.4)	71.0 (68.3–75.4)	Training
	56.6 (54.5–58.8)	50.5 (48.7–53.1)	55.0 (52.9–57.5)	76.3 (73.6–80.9)	65.0 (62.6–68.1)	Test
Lwin = 64 OP = 50%	68.9 (65.9–70.8)	70.5 (68.8–72.6)	69.3 (66.9–71.2)	87.5 (80.1–90.0)	77.1 (72.3–79.3)	Training
	58.7 (56.1–60.9)	55.5 (53.2–58.1)	57.8 (55.3–60.2)	77.1 (75.9–81.3)	66.7 (64.5–69.7)	Test
Lwin = 64 OP = 75%	76.5 (74.1–78.1)	75.5 (73.5–79.6)	76.2 (73.9–78.6)	87.9 (84.8–88.4)	81.8 (79.1–83.0)	Training
	59.2 (56.9–62.2)	56.0 (53.2–59.2)	58.3 (55.8–61.3)	77.5 (74.2–78.1)	67.1 (64.4–69.2)	Test
Lwin = 128 OP = 75%	88.3 (86.1–91.1)	**89.5 (87.1–92.3)**	**88.7 (86.5–91.2)**	**94.4 (90.9–99.2)**	91.2 (88.4–95.0)	Training
	72.6 (69.8–73.9)	68.4 (66.2–71.4)	71.5 (68.6–73.3)	*86.6 (80.5–89.1)*	79.0 (74.8–80.8)	Test
Lwin = 256 OP = 75%	**88.8 (86.2–91.2)**	87.5 (85.3–89.6)	88.4 (85.9–90.9)	94.1 (92.1–97.4)	91.4 (89.1–94.2)	Training
	79.7 (77.3–82.7)	*70.5 (68.1–72.5)*	*76.6 (74.1–79.4)*	84.2 (82.0–86.3)	*81.9 (79.6–84.5)*	Test

The optimum values of performance criteria at stage of training and test are highlighted in bold and italics font respectively



(a) training stage

(b) test stage

Fig. 7 The confusion matrices with spectrogram feature

Precision reflects how reliable the model is in classifying samples as positive. F1_score is defined as the harmonic mean of precision and sensitivity, which combines the two metrics into a single metric, and works well on imbalance data particularly. These metrics are used most widely to characterize the performance of a classifier, such as CNN and SVM.

In the following, the process of training and test will be performed 100 times randomly, and then the mean value and 95% confidence intervals (CIs) for these metrics will be derived.

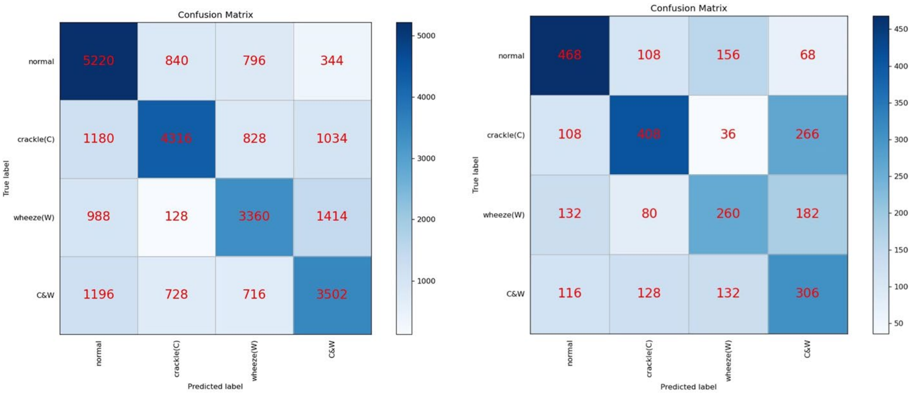
4 Results

The performance criteria and confusion matrices of the CNN are shown in Table 7 and Fig. 7, respectively, when spectrogram is used as feature. For simplicity, only five typical parameter combinations are presented in it. Under these combinations, the criteria of sensitivity, specificity, accuracy, precision and F1_score are shown. The confusion matrices are exhibited only under the parameter combination of Lwin = 256, OP = 75%.

Table 8 Performance criteria with MFCCs feature

Parameter combination	Performance criteria (%)					Stage
	Sensitivity (CI)	Specificity (CI)	Accuracy (CI)	Precision (CI)	F1_score (CI)	
Lwin = 64 OP = 25%	59.3 (57.1–61.5)	59.1 (56.7–61.1)	59.2 (56.9–61.3)	59.2 (56.9–61.3)	59.2 (57.0–61.4)	Training
	53.3 (50.1–55.2)	49.5 (47.2–52.0)	52.3 (49.2–54.7)	74.7 (67.8–86.1)	62.2 (57.6–67.3)	Test
Lwin = 64 OP = 50%	62.6 (59.3–64.7)	62.1 (60.1–65.2)	62.5 (59.6–64.8)	86.9 (71.2–88.1)	72.8 (64.7–74.6)	Training
	55.8 (54.1–58.1)	49.5 (47.2–51.8)	54.1 (52.3–56.6)	74.9 (74.4–79.4)	64.0 (62.6–67.1)	Test
Lwin = 64 OP = 75%	65.0 (62.9–67.3)	69.1 (67.3–72.1)	66.1 (64.2–68.2)	85.2 (82.1–91.3)	73.7 (71.2–77.5)	Training
	57.4 (54.8–59.6)	50.1 (48.2–52.3)	55.4 (52.6–57.7)	75.3 (67.9–78.0)	65.1 (60.7–67.6)	Test
Lwin = 128 OP = 75%	73.7 (70.9–75.6)	69.9 (67.1–72.3)	72.2 (69.2–74.4)	79.0 (72.7–82.7)	76.2 (71.8–79.0)	Training
	64.3 (62.1–66.5)	*65.0 (62.9–67.4)*	64.5 (62.5–66.7)	82.1 (62.6–87.7)	72.1 (62.3–75.6)	Test
Lwin = 256 OP = 75%	**76.7 (74.2–78.9)**	**73.1 (70.0–75.3)**	**75.5 (72.6–77.8)**	**85.1 (80.1–87.9)**	**80.7 (77.0–83.2)**	Training
	72.1 (69.8–74.9)	64.5 (62.1–66.5)	*69.3 (66.9–72.2)*	*77.7 (75.3–82.5)*	*74.8 (72.4–78.5)*	Test

The optimum values of performance criteria at stage of training and test are highlighted in bold and italics font respectively



(a) training stage

(b) test stage

Fig. 8 The confusion matrices with MFCCs feature

The performance criteria and confusion matrices of the CNN are shown in Table 8 and Fig. 8, respectively, when MFCCs are used as feature. The same parameter combinations as in Table 7 are presented in the following table.

The above results show that the larger Lwin and OP, and the better the performance for lung sounds classification. The larger Lwin, the greater the frequency resolution. And the larger the OP, the greater the time resolution. It means that a larger frequency and time resolution is beneficial to improve the classification performance of the CNN. However, when the frequency resolution reaches a specific value, the improvement on classification performance is no longer significant. When the OP is kept at 75% and the window length increases from 128 to 256, either at training or test stage, no significant improvement on classification performance has been found at all, and some performance criteria have even begun to decrease. In addition, the larger Lwin, the higher the requirements for computation and storage resources for FFT. There should be a compromise between the requirements of performance improvement and computation or storage capacity, and select appropriate Lwin and OP. From Tables 7 and 8, it can be concluded that Lwin 128 and OP 75% are a relatively optimum parameter combination.

By comparing the data between Tables 7 and 8, and Figs. 7 and 8, it can be seen that under the same parameter combination, at both training and test stage, the performance criteria of the CNN with spectrogram feature are significantly better than the criteria of the CNN with MFCCs feature. The reason may lie in the fact that the resolution of MFCCs is weaker than that of spectrogram. When the machine learning models, such as SVM with MFCCs feature are used to classify lung sounds, they may show advantages over DLMs. However, as a hand-crafted feature, MFCCs may be more biased in the generation process, resulting in incomplete feature representation. This might be one of the reasons for its slightly worse performance in DLMs.

The receiver operating characteristic (ROC) curves are shown in Fig. 9, under four different parameter combinations. As shown in it, the best performance is achieved under the parameter combination of Lwin 256, OP 75% and spectrogram feature. However, under the spectrogram feature and OP 75%, there is no significant improvement on performance compared to the performance with Lwin 128. The AUCs (area under ROC curve) are 0.93 and 0.95 under the two settings, showing no difference almost.

In summary, it can be seen that with the sampling frequency of 8 kHz, the parameter combination of Lwin 128, OP 75% and the spectrogram feature can achieve superior performance to the same combination but MFCCs feature. With this combination, the time complexity of the CNN is $O(1241 * 59 * 7 * 7 * 64) = O(229614784)$ without considering the activation layer.

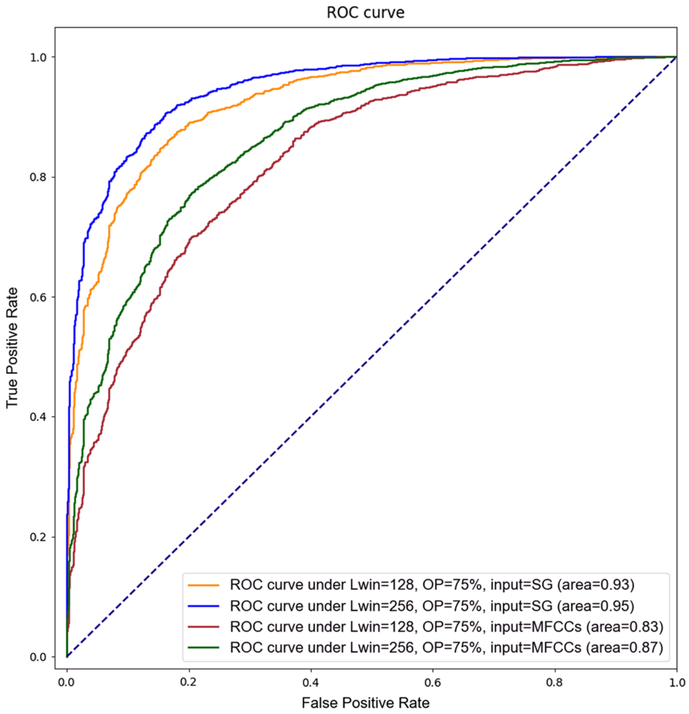


Fig. 9 ROC curves under four different parameter combinations

5 Discussion

The results show that using spectrogram feature input will achieve better performance than using MFCCs feature. And under the same feature the classification performance may be affected by feature-related parameters heavily. Generally speaking, the longer the frame length and the larger the overlap percent of two successive frames, the better the classification performance.

The spectrogram is a visual representation of the spectrum of frequencies of a signal as it varies with time. When applied to an audio signal, spectrogram is sometimes called voiceprint or voicegram. It has been used as the indicator of speaker and applied to speaker recognition. And it has been accepted as a widely used feature to classify normal/abnormal lung sounds. For a segment of lung sound, the larger the frame length, the higher the frequency resolution, and but the lower the time resolution. So we must make a compromise between the two resolutions. From the experimental results, it can be concluded that frequency resolution contributes more to the classification performance than the counterpart does. The reason may lie in the fact that the lower frequency resolution will lead to a coarse voiceprint to some extent. It is natural to say that we cannot achieve more accurate from some coarse voiceprints.

MFCCs are commonly used as features in speech recognition systems and work relatively well. Unfortunately, they do not get a score high enough as expected in this work. Because the maximum order of MFCCs is fixed, the granularity of MFCCs feature is limited. The difference of features of the four types of lung sounds may not be represented significantly. So there exists difficulty for the CNN to classify lung sounds accurately.

We make a comparative study with similar up-to-date works. In order to ensure comparability, comparative study was limited to the similar works using spectrogram-like features and ICBHI 2017 dataset. The number of classes varies between these classifiers, such as 2-types (healthy/non-healthy, normal/abnormal), 3-types (wheeze/crackle/normal, healthy/non-chronic/chronic diseases, crackle/rhonchi/normal), 4-types (normal/crackle/wheeze/crackle and wheeze) and 6-types (healthy/bronchiectasis/Bronchiolitis/COPD/Pneumonia/URTI). Among them, only the works performing 4 types of classification were selected for comparison. The performance metrics for these works along with the proposed one are provided in Table 9. Among these works, our proposed classifier with recommended parameters can achieve relatively better performance.

Table 9 Performance comparison between the proposed work (Lwin = 128, OP = 75%, SG feature) and state-of-the-art works as quadruple classifiers based on ICBHI 2017

Works	Sen (%)	Spe (%)	Pre (%)	Acc (%)	F1_score (%)
Petmezas et al. [12]	52.78	84.26	–	76.39	68.52
Acharya et al. [28]	48.63	84.4	–	–	–
Jayalakshmy et al. [32]	–	–	–	**92.5**	–
Asatani et al. [33]	63	83	–	–	72
Rishabh et al. [38]	67.22	82.87	–	–	75.04
Jakovljević et al. [41]	–	–	–	–	39.56
proposed work	**88.3**	87.5	**94.4**	88.7	**91.2**

The optimum values of performance criteria are highlighted in bold

Acc accuracy, *Sen* sensitivity or recall, *Pre* precision, *Spe* specificity

Sometimes, we focus on the responsible section of the image rather than classification accuracy, especially in clinic analysis [51]. Unlike medical images such as X-rays and CT images, lung sounds are acoustic signal in. Physicians are trained to make diagnosis by auscultation. In order to use CNNs for lung sounds classification, we must convert acoustic signals into spectrogram-like images. These images are intermediate results and cannot be shown to physicians directly. Even the responsible section is marked by some method such as GradCam, it has little significance in guiding the diagnosis of lung diseases for the difficulty in perception [25]. So this part is not included in this paper.

6 Conclusions and outlook

The performance of the deep learning model, namely CNN, under different parameter combinations and two types of features are investigated in detail by experiments. Combined with the two types of features, two parameters of frame length and overlap percentage (OP) of successive frames are emphasized. The spectrogram and MFCCs of lung sounds are used as features to the CNN, respectively. The training and test results show that there is significant difference on performance under varied parameter combinations and features. From the results, we can see that OP is a performance sensitive parameter. The higher OP, the better overall performance. Meanwhile, more computation and storage resource is needed for higher OP. So OP is restricted to maximum of 75% for practical purpose. We fix the sampling frequency to 8 kHz without loss of important characteristics of lung sounds, because the maximum frequency of lung sounds is not above 3 kHz. When the frame size increases to 128 or more, the improvement on the performance is slight. We can hardly see significant difference between the performance metrics between the CNN with frame size of 128 and 256. However, when the frame size decreases from 128 to 64 or even less, the performance of the CNN degrades rapidly. It can also see that the CNN with spectrogram feature shows more excellent performance than the one with MFCCs feature under the same parameter combination. So it is concluded that frame size 128, OP 75% and spectrogram input is the optimum parameter setting, under which a compromise between performance and resources requirement can be reached.

In future, on one hand, we will evaluate the performance by considering more parameters or another deep learning model. For example, some other features and data augmentation methods would be tried. The background noise recorded from the hospital will be inserted in the audio samples, instead of white noise. And the log-scale spectrogram is another choice for the feature. In addition, we could compare the performance of CNN with another deep learning model, such as RNN. We will also try to combine more open respiratory databases, such as RespiratoryDatabase@TR [20] for CNN training and test. On the other hand, the CNN is running on the GPU platform presently. For practical purpose, the model should be simplified in order to be transferred to the embedded platform. The most ideal implementation is to train and test a lightweight CNN [30, 51] and run it on a electronic stethoscope, which will help the physician to distinguish normal/abnormal lung conditions as quickly as possible.

Acknowledgements
We would like to thank two anonymous reviewers and the editor for their comments.

Author contributions

Zhaoping Wang and Zhiqiang Sun made substantial contributions to conception and design. Zhaoping Wang performed data acquisition and validation for this study. Zhiqiang Sun implemented the CNN modeling and training with lung sounds. Zhaoping Wang edited the manuscript, and both of them contributed to elaboration and redaction of the final manuscript.

Availability of data and materials

The ICBHI 2017 respiratory sound database is used to train and test the deep learning model. It was originally compiled to support the scientific challenge organized at Int. Conf. on Biomedical Health Informatics - ICBHI 2017. The current version of this database is made publicly available for research. We can download the raw data of lung sounds on https://bhichallenge.med.auth.gr/sites/default/files/ICBHI_final_database/ICBHI_final_database.zip. We have also uploaded the raw data file (*.zip file) to Harvard Dataverse and shared it via the https://doi.org/10.7910/DVN/HT6PKI. It is the only dataset supporting this paper. The final results and data generated or analysed during this study are included in this published article. The intermediate data are available from the authors upon reasonable request.

Declarations

Competing interests

The authors declare no competing interests.

Received: 16 October 2023 Accepted: 5 April 2024
Published online: 15 April 2024

References

1. World Health Statistics 2022, *Technical Report* (World Health Organization, Geneva, 2022)
2. F. Demir, A. Sengur, V. Bajaj, Convolutional neural networks based efficient approach for classification of lung diseases. Health Inf. Sci. Syst. **8**, 4 (2019). https://doi.org/10.1007/s13755-019-0091-3
3. A. Mondal, P. Banerjee, H. Tang, A novel feature extraction technique for pulmonary sound analysis based on EMD. Comput. Methods Programs Biomed. **159**, 199–209 (2018). https://doi.org/10.1016/j.cmpb.2018.03.016
4. Q.-H. He, B. Yu, X. Hong, B. Lv, T. Liu, J. Ran, Y.-T. Bi, An improved lung sound de-noising method by wavelet packet transform with PSO-based threshold selection. Intell. Autom. Soft Comput. **24**(2), 223–230 (2018). https://doi.org/10.1080/10798587.2016.1261957
5. M. Gronnesby, Automated lung sound analysis. Master's thesis, The Arctic University of Norway, Norway (2016)
6. S. İçer, S. Gengeç, Classification and analysis of non-stationary characteristics of crackle and rhonchus lung adventitious sounds. Digit. Signal Process. **28**, 18–27 (2014). https://doi.org/10.1016/j.dsp.2014.02.001
7. R. Naves, B.H.G. Barbosa, D.D. Ferreira, Classification of lung sounds using higher-order statistics: a divide-and-conquer approach. Comput. Methods Programs Biomed. **129**, 12–20 (2016). https://doi.org/10.1016/j.cmpb.2016.02.013
8. R. Palaniappan, K. Sundaraj, N.U. Ahamed, Machine learning in lung sound analysis: a systematic review. Biocybern. Biomed. Eng. **33**(3), 129–135 (2013). https://doi.org/10.1016/j.bbe.2013.07.001
9. H. Zhu, J. Lai, B. Liu, Z. Wen, Y. Xiong, H. Li, Y. Zhou, Q. Fu, G. Yu, X. Yan, X. Yang, J. Zhang, C. Wang, H. Zeng, Automatic pulmonary auscultation grading diagnosis of coronavirus disease 2019 in China with artificial intelligence algorithms: a cohort study. Comput. Methods Programs Biomed. **213**, 106500 (2022). https://doi.org/10.1016/j.cmpb.2021.106500
10. Y. Kim, Y. Hyon, S.S. Jung, S. Lee, G. Yoo, C. Chung, T. Ha, Respiratory sound classification for crackles, wheezes, and rhonchi in the clinical field using deep learning. Sci. Rep. **11**(1), 17186 (2021). https://doi.org/10.1038/s41598-021-96724-7
11. M. Aykanat, Ö. Kılıç, B. Kurt, S. Saryal, Classification of lung sounds using convolutional neural networks. EURASIP J. Image Video Process. **2017**(1), 1–9 (2017). https://doi.org/10.1186/s13640-017-0213-2
12. G. Petmezas, G.-A. Cheimariotis, L. Stefanopoulos, B. Rocha, R.P. Paiva, A.K. Katsaggelos, N. Maglaveras, Automated lung sound classification using a hybrid CNN-LSTM network and focal loss function. Sensors **22**(3), 1232 (2022). https://doi.org/10.3390/s22031232
13. Y. Ma, X. Xu, Y. Li, LungRN+NL: an improved adventitious lung sound classification using non-local block ResNet neural network with Mixup data augmentation, in *Proceedings of the Interspeech 2020* (2020), pp. 2902–2906. https://doi.org/10.21437/Interspeech.2020-2487
14. B.M. Rocha, D. Pessoa, A. Marques, P. Carvalho, R.P. Paiva, Automatic classification of adventitious respiratory sounds: A (un)solved problem? Sensors **21**(1), 57 (2021). https://doi.org/10.3390/s21010057
15. K. Tunyasuvunakool, J. Adler, Z. Wu, T. Green, M. Zielinski, A. Žídek, A. Bridgland, A. Cowie, C. Meyer, A. Laydon et al., Highly accurate protein structure prediction for the human proteome. Nature **596**(7873), 590–596 (2021). https://doi.org/10.1038/s41586-021-03828-1
16. A. Fourcade, R.H. Khonsari, Deep learning in medical image analysis: a third eye for doctors. J. Stomatol. Oral Maxillofac. Surg. **120**(4), 279–288 (2019). https://doi.org/10.1016/j.jormas.2019.06.002
17. B.M. Rocha, D. Filos, L. Mendes, I. Vogiatzis, E. Perantoni, E. Kaimakamis, P. Natsiavas, A. Oliveira, C. Jácome, A. Marques, R.P. Paiva, I. Chouvarda, P. Carvalho, N. Maglaveras, A respiratory sound database for the development of automated classification, in *Precision Medicine Powered by pHealth and Connected Health*. ed. by N. Maglaveras, I. Chouvarda, P. Carvalho (Springer, Singapore, 2017), pp.33–37. https://doi.org/10.1007/978-981-10-7419-6_6
18. Q. Zhang, J. Zhang, J. Yuan, H. Huang, Y. Zhang, B. Zhang, G. Lv, S. Lin, N. Wang, X. Liu, M. Tang, Y. Wang, H. Ma, L. Liu, S. Yuan, H. Zhou, J. Zhao, Y. Li, Y. Yin, L. Zhao, G. Wang, Y. Lian, SPRSound: open-source SJTU paediatric respiratory

sound database. IEEE Trans. Biomed. Circuits Syst. **16**(5), 867–881 (2022). https://doi.org/10.1109/TBCAS.2022.3204910

19. H. Fu-Shun, H. Shang-Ran, H. Chien-Wen, C. Yuan-Ren, C. Chun-Chieh, H. Jack, C. Chung-Wei, L. Feipei, A progressively expanded database for automated lung sound analysis: an update. Appl. Sci. **12**(15), 7623 (2022). https://doi.org/10.3390/app12157623
20. G. Altan, Y. Kutlu, Y. Garbi, A.O. Pekmezci, S. Nural, Multimedia respiratory database (RespiratoryDatabase@TR): auscultation sounds and chest X-rays (2021). https://doi.org/10.48550/arXiv.2101.10946
21. J.J. Ward, R.A.L.E lung sounds 3.1. Respir. Care **50**, 1385–1388 (2005)
22. A. Theodore, S. Vladimir, F. Evgeny, K. Natalia, F. Gregory, Application of deep learning for bronchial asthma diagnostics using respiratory sound recordings. Peer J. Comput. Sci. **9**, 1173 (2023). https://doi.org/10.7717/peerj-cs.1173
23. M. Fraiwan, L. Fraiwan, M. Alkhodari, O. Hassanin, Recognition of pulmonary diseases from lung sounds using convolutional neural networks and long short-term memory. J. Ambient Intell. Hum. Comput. **13**, 4759–4771 (2022). https://doi.org/10.1007/s12652-021-03184-y
24. B.A. Tessema, H.D. Nemomssa, G.L. Simegn, Acquisition and classification of lung sounds for improving the efficacy of auscultation diagnosis of pulmonary diseases. Med. Devices Evid. Res. **15**, 89–102 (2022). https://doi.org/10.2147/MDER.S362407
25. Y. Choi, H. Lee, Interpretation of lung disease classification with light attention connected module. Biomed. Signal Process. Control **84**, 104695 (2023). https://doi.org/10.1016/j.bspc.2023.104695
26. G. Altan, Y. Kutlu, A. Gökçen, Chronic obstructive pulmonary disease severity analysis using deep learning on multichannel lung sounds. Turk. J. Electr. Eng. Comput. Sci. **28**(5), 2979–2996 (2020). https://doi.org/10.3906/elk-2004-68
27. A. Roy, U. Satija, A novel melspectrogram snippet representation learning framework for severity detection of chronic obstructive pulmonary diseases. IEEE Trans. Instrum. Meas. **72**(4003311), 1–11 (2023). https://doi.org/10.1109/TIM.2023.3256468
28. J. Acharya, A. Basu, Deep neural network for respiratory sound classification in wearable devices enabled by patient specific model tuning. IEEE Trans. Biomed. Circuits Syst. **14**(3), 535–544 (2020). https://doi.org/10.1109/TBCAS.2020.2981172
29. H. Chen, X. Yuan, Z. Pei, M. Li, J. Li, Triple-classification of respiratory sounds using optimized s-transform and deep residual networks. IEEE Access **7**, 32845–32852 (2019). https://doi.org/10.1109/ACCESS.2019.2903859
30. S.B. Shuvo, S.N. Ali, S.I. Swapnil, T. Hasan, M.I.H. Bhuiyan, A lightweight CNN model for detecting respiratory diseases from lung auscultation sounds using EMD-CWT-based hybrid scalogram. IEEE J. Biomed. Health Inform. **25**(7), 2595–2603 (2021). https://doi.org/10.1109/JBHI.2020.3048006
31. F. Cinyol, U. Baysal, D. Köksal, E. Babaoğlu, S.S. Ulaşlı, Incorporating support vector machine to the classification of respiratory sounds by convolutional neural network. Biomed. Signal Process. Control **79**, 104093 (2023). https://doi.org/10.1016/j.bspc.2022.104093
32. S. Jayalakshmy, G.F. Sudha, Conditional GAN based augmentation for predictive modeling of respiratory signals. Comput. Biol. Med. **138**, 104930 (2021). https://doi.org/10.1016/j.compbiomed.2021.104930
33. N. Asatani, T. Kamiya, S. Mabu, S. Kido, Classification of respiratory sounds using improved convolutional recurrent neural network. Comput. Electr. Eng. **94**, 107367 (2021). https://doi.org/10.1016/j.compeleceng.2021.107367
34. D. Perna, Convolutional neural networks learning from respiratory data, in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* (2018), pp. 2109–2113. https://doi.org/10.1109/BIBM.2018.8621273
35. A. Dhavala, A. Ahmed, R. Periyasamy, D. Joshi, An MFCC features-driven subject-independent convolution neural network for detection of chronic and non-chronic pulmonary diseases, in *2022 3rd International Conference for Emerging Technology (INCET)* (2022), pp. 1–9. https://doi.org/10.1109/INCET54531.2022.9824677
36. A.A. Saraiva, D.B.S. Santos, A.A. Francisco, J.V.M. Sousa, N.M.F. Ferreira, S. Soares, A. Valente, Classification of respiratory sounds with convolutional neural network, in *Proceedings of the 13th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2020)—BIOINFORMATICS, INSTICC* (SciTePress, Valletta, 2020), pp. 138–144. https://doi.org/10.5220/0008965101380144
37. Z. Tariq, S.K. Shah, Y. Lee, Feature-based fusion using CNN for lung and heart sound classification. Sensors **22**(4), 1521 (2022). https://doi.org/10.3390/s22041521
38. D. Rishabh Kumar, Multi spectral feature extraction to improve lung sound classification using CNN, in *2023 10th International Conference on Signal Processing and Integrated Networks (SPIN)* (2023), pp. 186–191. https://doi.org/10.1109/SPIN57001.2023.10116295
39. B. Tasar, O. Yaman, T. Tuncer, Accurate respiratory sound classification model based on piccolo pattern. Appl. Acoust. **188**, 108589 (2022). https://doi.org/10.1016/j.apacoust.2021.108589
40. B. Ari, O.F. Alçin, A. Şengür, A lung sound classification system based on data augmenting using ELM-wavelet-AE. Turk. J. Sci. Technol. **17**(1), 79–88 (2022). https://doi.org/10.55525/tjst.1063039
41. N. Jakovljević, T. Lončar-Turukalo, Hidden Markov model based respiratory sound classification, in *Precision Medicine Powered by pHealth and Connected Health*. ed. by N. Maglaveras, I. Chouvarda, P. Carvalho (Springer, Singapore, 2018), pp.39–43. https://doi.org/10.1007/978-981-10-7419-6_7
42. Y. Hui, Z. Jing, Q. Zhaoyu, L. Dongyi, C. Zhen, G. Chengxiang, S. Jinglai, Z. Xiaoyun, Diagnosis model of chronic obstructive pulmonary disease based on deep learning. Chin. J. Biomed. Eng. **41**(5), 558 (2022). https://doi.org/10.3969/j.issn.0258-8021.2022.05.005
43. G. Altan, Y. Kutlu, A. Pekmezci, S. Nural, Deep learning with 3d-second order difference plot on respiratory sounds. Biomed. Signal Process. Control **45**, 58–69 (2018). https://doi.org/10.1016/j.bspc.2018.05.014
44. G. Altan, Y. Kutlu, N. Allahverdi, Deep learning on computerized analysis of chronic obstructive pulmonary disease. IEEE J. Biomed. Health Inform. **24**(5), 1344–1350 (2020). https://doi.org/10.1109/JBHI.2019.2931395
45. A. Gökçen, Computer-aided diagnosis system for chronic obstructive pulmonary disease using empirical wavelet transform on auscultation sounds. Comput. J. **64**(11), 1775–1783 (2021). https://doi.org/10.1093/comjnl/bxaa191
46. A.M. Alqudah, S. Qazan, Y.M. Obeidat, Deep learning models for detecting respiratory pathologies from raw lung auscultation sounds. Soft Comput. **26**(24), 13405–13429 (2022)

47. J. Salamon, J.P. Bello, Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Process. Lett. **24**(3), 279–283 (2017). https://doi.org/10.1109/LSP.2017.2657381
48. R. Zulfiqar, F. Majeed, R. Irfan, H.T. Rauf, E. Benkhelifa, A.N. Belkacem, Abnormal respiratory sounds classification using deep CNN through artificial noise addition. Front. Med. **8**, 714811 (2021). https://doi.org/10.3389/fmed.2021.714811
49. A. Amjad, L. Khan, H.-T. Chang, Data augmentation and deep neural networks for the classification of Pakistani racial speakers recognition. PeerJ Comput. Sci. **8**, 1053 (2022). https://doi.org/10.7717/peerj-cs.1053
50. B.M. Rocha, D. Pessoa, A. Marques, P. Carvalho, R.P. Paiva, Influence of event duration on automatic wheeze classification, in *2020 25th International Conference on Pattern Recognition (ICPR)* (IEEE, 2021), pp. 7462–7469. https://doi.org/10.1109/ICPR48806.2021.9412226
51. G. Altan, DeepOCT: an explainable deep learning architecture to analyze macular edema on OCT images. Eng. Sci. Technol. Int. J. **34**, 101091 (2022). https://doi.org/10.1016/j.jestch.2021.101091

Publisher's Note