Artificial Intelligence in Medicine 88 (2018) 58–69

Contents lists available at ScienceDirect

Artificial Intelligence in Medicine

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

Lung sounds classification using convolutional neural networks

Dalal Bardou [a,*], Kun Zhang [a,*], Sayed Mohammad Ahmad [b]

[a] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China
[b] Lareb Technologies, India

ARTICLE INFO

Article history:
Received 5 May 2017
Received in revised form 18 April 2018
Accepted 23 April 2018

Keywords:
Convolutional neural network
Lung sounds classification
Handcrafted features extraction
Deep learning
Models ensembling
Support vector machines

ABSTRACT

Lung sounds convey relevant information related to pulmonary disorders, and to evaluate patients with pulmonary conditions, the physician or the doctor uses the traditional auscultation technique. However, this technique suffers from limitations. For example, if the physician is not well trained, this may lead to a wrong diagnosis. Moreover, lung sounds are non-stationary, complicating the tasks of analysis, recognition, and distinction. This is why developing automatic recognition systems can help to deal with these limitations. In this paper, we compare three machine learning approaches for lung sounds classification. The first two approaches are based on the extraction of a set of handcrafted features trained by three different classifiers (support vector machines, k-nearest neighbor, and Gaussian mixture models) while the third approach is based on the design of convolutional neural networks (CNN). In the first approach, we extracted the 12 MFCC coefficients from the audio files then calculated six MFCCs statistics. We also experimented normalization using zero mean and unity variance to enhance accuracy. In the second approach, the local binary pattern (LBP) features are extracted from the visual representation of the audio files (spectrograms). The features are normalized using whitening. The dataset used in this work consists of seven classes (normal, coarse crackle, fine crackle, monophonic wheeze, polyphonic wheeze, squawk, and stridor). We have also experimentally tested dataset augmentation techniques on the spectrograms to enhance the ultimate accuracy of the CNN. The results show that CNN outperformed the handcrafted feature based classifiers.

1. Introduction

Lung sound characteristics and their diagnoses form an indispensable part of pulmonary pathology [1,2]. Auscultation is a technique whereby physicians evaluate and diagnose patients with pulmonary conditions by using a stethoscope. It is known to be inexpensive, non-invasive, and safe, besides taking less time for diagnosis [1,2]. It also provides much information about the respiratory organ and the signs of the diseases that affect it [3]. However, if not done by a well-trained physician, this may lead to wrong diagnosis.

Lung sounds being non-stationary signals, it is both difficult to analyze and hard to distinguish them with traditional auscultation methods. Thus, the use of an electronic stethoscope, coupled with a pattern recognition system, helps to overcome the limitations of traditional auscultation; and provides an efficient method for clinical diagnosis [4,5].

Lung sounds can be divided into two categories: normal breathing sounds or adventitious breathing sounds. Normal breathing sounds are heard when no respiratory disorders exist and adventitious sounds are heard when a respiratory disorder does exist [6,7]. Normal respiratory sounds consist of tracheal, bronchial and bronchovesicular sounds. On the chest wall, a normal respiratory sound is characterized by a low noise during inspiration, and a hardly audible noise during expiration. On the trachea, a normal respiratory sound is characterized by a broader spectrum of noise, such as a noise containing higher-frequency components, which is audible during both the inspiratory and expiratory phase [8]. An adventitious sound is an additional respiratory sound that is superimposed onto normal breath sounds. These can be continuous, like wheezes, or discontinuous, like crackles. Some of them, like squawks, have both characteristics. The presence of such sounds usually indicates a pulmonary disorder [8].

Crackles are discontinuous and explosive adventitious sounds. They appear much more during the inspiratory phase. They are characterized by their specific waveform, their duration, and their location in the respiratory cycle. A crackle can be characterized by its total duration: fine crackles have a short duration and coarse crackles have a long duration [8]. Fine crackles are present in high frequencies. Crackle occurs in many diseases, such as congestive

* Corresponding authors.
E-mail addresses: dalal.bardou@njust.edu.cn (D. Bardou), zhangkun@njust.edu.cn (K. Zhang).

https://doi.org/10.1016/j.artmed.2018.04.008
0933-3657/

heart failure, pneumonia, bronchiectasis, pulmonary fibrosis, and chronic diffuse parenchymal lung disease. The American Thoracic Society [9] have defined fine crackles as having an initial deflection width (IDW) of 0.7 ms and 2 cycle durations (2CD) of 5 ms, and coarse crackles as having an IDW of 1.5 ms and 2CD of 10 ms. Another group has defined fine crackles as having a 2CD < 10 ms, and coarse crackles as having a 2CD > 10 ms [10]. Additionally, the frequency spectrum of crackles is between 200 Hz and 2000 Hz [11,12].

Wheezes are high-pitched continuous adventitious sounds. A wheeze can be monophonic if it contains a single frequency, or polyphonic if several frequencies are simultaneously perceived [11]. Rhonchi are low-pitched, continuous sounds, and they are characterized by a dominant frequency of about 200 Hz or less [5]. The diseases associated with wheezing sounds are asthma, pneumonia, and bronchitis.

There are many other categories of lung sounds, including pleural rub, stridor, and squawks. A pleural rub is the characteristic sound produced when inflamed pleural surfaces rub together during respiration; a stridor is a very loud wheeze; a squawk is a short inspiratory wheeze [13].

In this paper, we have proposed three classification approaches to classifying seven types of lung sounds. The lung sounds consist of normal sounds, coarse crackle, fine crackle, monophonic wheeze, polyphonic wheeze, stridor, and squawk. The objectives of our study are: **(a)** to compare the handcrafted features based classification methods and the convolutional neural networks, **(b)** to test the power and the suitability of the convolutional neural networks to address lung sounds classification task, **(c)** to put to the test dataset normalization and augmentations influence on the performances, and **(d)**, to compare between handcrafted features extracted directly from audio files and the ones extracted from the visual representation of audio files (spectrograms).

The rest of the paper is organized as follows: in Section 2, we give an overview of the previous studies related to lung sounds classification using handcrafted features based classification methods, and the recent convolutional neural networks applications related to medical sounds classification as well. In Section 3, we give information about the original data and explained how the data set is constructed and preprocessed, while Section 4 is dedicated to the data augmentation techniques applied to spectrograms. In Section 5, we give the topology of the proposed CNN. In Section 6, we describe the handcrafted features-based classification, the classifiers, as well as the implementations settings. Finally, in Section 7 and Section 8, performances and experimental results are provided, and a comparison with the literature is made.

2. Related work

With the onset of pattern recognition and artificial intelligence, many feature-based approaches have been proposed, to develop automatic systems for the classification of different lungs sounds. The support vector machine (SVM) is known to be a promising method, and so it has been used to address this task. In [4], the authors used the frequency ratio of power spectral density (PSD) values, and the Hilbert-Huang transform (HHT) features, to distinguish between normal lung sounds, crackles, and rhonchus. The SVM classifier achieved an accuracy of above 90%. In [14], time-frequency (TF) and time-scale (TS) analysis are proposed for the detection of pulmonary crackles. In the feature extraction step, the frequency characteristics of crackles, using TF and TS analysis, are extracted from both the non-pre-processed and pre-processed signals. For pre-processing, DTCWT is applied, with the aim of removing the frequency bands that do not contain crackle information. In classification step, k-Nearest Neighbors (k-NN), SVM, and multilayer perceptron are used to classify crackling and non-crackling sounds. The SVM classifier achieved the best result, obtaining a classification accuracy of 97.5%. In [15], SVM is used to classify respiratory sounds, ranging from normal to continuous adventitious, including wheezing, stridor, and rhonchi. The removed features consisted of instantaneous kurtosis, discriminating function, and entropy. The optimally achieved classification accuracy was between 97.7% and 98.8%. In [16], another approach is proposed for the classification of normal and continuous adventitious signals, although only wheeze signals were used as continuous adventitious signals. The mel-frequency cepstral coefficients (MFCCs) are used for feature extraction, the gaussian mixture model (GMM) is used to classify the signals, and the achieved accuracy was 94.2%. In [17], a technique to obtain the time-frequency representation (TFR) of thoracic sounds is proposed. Using TF patterns, the authors assessed the performance of the TFRs for the heart, adventitious, and normal lung sounds. After simulations, they concluded that the best TFR performance was achieved by the Hilbert–Huang spectrum (HHS). In [5], the classification of lung sounds using higher order statistics (HOS) is proposed. In feature extraction step, HOS is used to extract features (second, third, and fourth order cumulants) from five types of lung sounds (normal, coarse crackle, fine crackle, monophonic and polyphonic wheezes). Genetic algorithms and Fisher's discriminant ratio are used to reduce dimensionality, and for classification, k-NN, and naive Bayes classifiers are used to classify lung sound events in a tree-based system. The classifier accuracy was 98.1% accuracy on training data, and 94.6% on validation data. In [18], the authors used MFCC features along with artificial neural network (ANN) to classify normal sounds, wheezes, and crackles. The classification achievement performance was 75% for crackles, 100% for wheezes and 80% for normal. The authors, in comparison with previous studies in [19,20], concluded that GMM is 15% more accurate than ANN for crackle classification. In [21], the authors proposed a method for the separation and classification of crackles from normal respiratory sounds, using GMM. This work consists of four steps: preprocessing, feature extraction, feature selection and classification. In the preprocessing step, a band-pass filter is used for background noises reduction, and then three spatial-temporal features, namely pitch, energy, and spectrogram, were extracted. After the feature selection step, the final features are trained using GMM. The achieved accuracy was 97.56%. In [22], the authors proposed a novel attractor recurrent neural networks (ARNN) topology, based on the fuzzy functions (FFs-ARNN), for the classification of lung abnormalities. The respiratory sounds are modelled using an ARNN and a FFs-ARNN, and to evaluate their performances, recurrent quantification analysis (RQA) was used. The best accuracy was 91%, which was achieved using FFs-ARNN with RQA features. While in [23], the authors propose a new recurrent fuzzy filter, based on a pipelined Takagi–Sugeno–Kang for a real-time separation of discontinuous adventitious sounds (DAS) and vesicular sounds (VS), in [24], an orthogonal least squares-based fuzzy filter (OLS-FF) is proposed for the same task. Two fuzzy inference systems are used in parallel to perform the task of adaptive separation, resulting in the OLS-FF. In [25], the authors proposed an automatic method for the elimination of vesicular sound from crackle signal, with minimal distortion of crackle parameters. After selecting a region of interest, distortion metric, based on the correlation between raw and filtered waveforms in that region, is defined. In [26], the authors proposed signal processing methodologies for the detection of crackles in audio files. After the extraction of a window of interest, based on fractal dimension and box filtering, the potential crackle is verified and validated, and crackle parameters are extracted and characterized.

The convolutional neural networks (CNN) can be regarded as a variant of the standard neural networks. Instead of using fully connected hidden layers, the CNNs introduce the structure of a special

Table 1
The lung sounds characteristics with the patients' information, audio file duration and the type of the disease.

Sound Type	Patient details	Inspiratory/expiratory	Associated Conditions	Disease	Duration
Normal Sound	Healthy 26-year-old man	both	Recorded with one stethoscope over the neck and another over the right lower lung on the back	None	20 s
	Healthy newborn baby girl on her second day of life	both	Was laying prone and the stethoscope was over her right posterior lower chest	None	10 s
	Healthy newborn baby boy on his third day of life	both	Recorded over the right lower chest	None	10 s
	Healthy six years old girl	both	Recorded over the right posterior lower chest	None	8.97 s
	Healthy 37-year-old male non-smoker	both	Recorded over the right lower chest	None	10 s
Monophonic Wheeze	17 years old boy	expiratory	Over the right anterior upper chest	Acute Asthma	20 s
Polyphonic Wheeze	79-year-old man	expiratory	Recorded both at his neck and over his right lower lung on the back	Chronic Obstructive Lung Disease	10 s
	Ten-month-old boy	Expiration	recorded over the right anterior upper chest	acute viral bronchiolitis	10 s
Stridor	2week old boy	Inspiration	Recorded Over the trachea	Laryngeal web	10 s
	55-year-old man	Both	recorded over the trachea	bronchogenic carcinoma	10 s
Squawk	78 years old woman	inspiratory	Recorded over the right posterior upper chest.	Interstitial pulmonary fibrosis	10 s
Fine Crackle	60 years old man	both	Over the right lower lung in the back	Interstitial pulmonary fibrosis	10 s
	60-year-old woman	Inspiration	recorded over the left lower lung on the back	Polymyositis and interstitial lung disease.	10 s
Coarse Crackle	15 years old boy	Early and mid-inspiration	Over the left posterior lower lung	Cystic fibrosis	10 s
	Nine-year-old boy	early and mid-inspiratory	recorded over the right lower lung on the back	cystic fibrosis With moderately severe lung disease.	10 s

network, which consists of so-called alternating convolution and pooling layers [27]. They have been first introduced for overcoming known problems of fully connected deep neural networks when handling high dimensionality structured inputs, such as images or speech [28]. CNNs have become state-of-the-art solutions for large scale object classification [29,31] and object detection tasks [30–32]. CNNs have been already applied to a variety of sounds problems. In [33], CNNs are used for the classification of environmental sounds. They are also used for classifying Hit-hat, snare and bass percussion sound samples with batch normalization in [34]. A system that performs automatic cough detection and that employs a wearable acoustic sensor and CNNs was presented in [35]. In [36], CNNs were used for normal/abnormal heart sound recordings classification, while in; [37] CNNs are combined with a classifier trained with time-frequency features.

3. Data creation

The respiratory sounds used in this work are the sounds of the R.A.L.E. ® (Respiration Acoustics Laboratory Environment) Lung Sounds 3.2 [38] which represents an educational program designed for students, educators, doctors, nurses and allied health professionals. The program originated at the respiratory acoustics laboratory of the University of Manitoba in Winnipeg, Canada. It offers more than 50 recordings of respiratory sounds in health and disease covering all age groups, and it includes quiz section for self-assessment. The quiz section presents an additional 24 clinical cases. The Health Sciences Communications Association has given for the production of R.A.L.E. ®, an Award of Merit, Computer-Based Materials [38].

Since the recordings can only be listened to within the program, we have contacted the owner to provide and allow using the data in research. The owner has provided **50 + BIN** files that correspond to the recordings found in the R.A.L.E Lung Sounds program and some additional audio files as well. The **RALEview** program can read the BIN files. **RALEview** also allows recordings to be exported as 16 bits WAV files. We have only selected the relevant files which include 15 audio files. The data consist of seven lung sounds types (normal, monophonic wheeze, polyphonic wheeze, coarse crackle, fine crackle, stridor, and squawks). Table 1 provides all the details: sound type, patient's age, associated conditions, type of disease and the duration of the recording. The normal lung sounds consists of tracheal, bronchial and bronchovesicular sounds. Sound channels are sampled at 10,240 **Hz** and contain **1024** points per segment. Each point is in the range −32,768 (−5.0 V) to + 32,767 (+5.0 V). In preprocessing step, and according to **CORSA** (the Computerized Respiratory Sound Analysis) guidelines [39], the signals were high-pass filtered at 7.5 Hz by using a first-order Butterworth filter to remove DC offset, they also low-pass filtered at 2.5 Hz by using 8th order Butterworth low-pass filter; and to reduce the effect of heart sounds, they are also filtered using a 4th order Butterworth bandpass filter (150–2000 Hz) [18,40]. The signals were divided into non-overlapped segments having each **40 ms**. The dataset is validated by a pulmonologist to ensure that we have a proper data and that the final results will be precise and accurate. Hamming windowing is also used to minimize the signal discontinuities at the beginning and end of each segment. The number of segments extracted from each class is provided in Table 2.

4. Data augmentation

The dataset augmentation techniques have been tested to increase the cardinality of the training set for all the classes, and to overcome the problem of overfitting. We have used two augmentation techniques applied to the spectrograms.

Table 2
Lung sounds segments numbers within each class.

Normal Sounds		Monophonic Wheeze		Polyphonic Wheeze		Squawk		Stridor		Coarse Crackle		Fine Crackle	
File	No. Segments	File	No. Segments	File	No. Segments	File	No. Segments	File	No. Segments	File	No. Segments	File	No. Segments
NO.1	345	NO.1	265	NO.1	58	NO.1	160	NO.1	98	NO.1	114	NO.1	114
NO.2	232	/	/	NO.2	89	/	/	NO.2	76	NO.2	87	NO.2	46
NO.3	253	/	/	/	/	/	/	/	/	/	/	/	/
NO.4	32	/	/	/	/	/	/	/	/	/	/	/	/
NO.5	172	/	/	/	/	/	/	/	/	/	/	/	/
Total	1034			Total	147			Total	174	Total	201	Total	160

- **Spectrogram Cropping:** A simple idea for data augmentation is random cropping. Each spectrogram was ten times randomly cropped.
- **Vocal tract length perturbation (VTLN):** Vocal tract length perturbation was firstly used for speaker normalization [41], and then for data augmentation [42,43] to improve speech recognition. VTLN has also been used as a data augmentation technique for deep neural networks (DNNs) and convolutional neural networks (CNNs) [44]. It uses the below formula [41] to wrap the frequency axis, and generate a random factor α; The initial frequency F is mapped to a new frequency F'.

$$G(f) = \begin{cases} \alpha f, & 0 \le f \le f_0 \\ \frac{f_{max} - \alpha f_0}{f_{max} - f_0}(f - f_0) + \alpha f_0, & f_0 \le f \le f_{max} \end{cases}$$

Where:

f_{max} : The maximum signal bandwidth.

f_0 : can be an empirically chosen frequency that falls above the highest significant formant in speech [41]. The warp factor is assumed to take value between 0.8 and 1.2. The range between 0.9 and 1.1 is used in this work to corrupt. The VTLN is applied to all the cropped images. The original spectrogram images have size of 750×256.

5. CNN classification

Convolutional neural network became a significant trend in machine learning, and it had much success in speech recognition, computer vision, and many other fields. In this work, we explored the power of the CNN in the classification of lung sounds. The topology of the proposed CNN is given below. The CNN has been implemented using BVLC Caffe [45] which is a popular and powerful framework of UC Berkeley. Caffe facilitates the design, the implementations of the CNNs, and it provides simple interfaces for both Matlab and Python. It also supports CPU and GPU training and provides an expressive architecture. The model and the optimization are defined in the configuration.

The topology of our proposed CNN is composed of five convolutional layers and two fully connected layers (Fig. 1) similar to AlexNet network [46] topology.

- 1st convolutional layer with filter size 7×7 and 64 feature maps.
- 2nd convolutional layer with filter size 5×5 and 128 feature maps.
- 3rd convolutional layer with filter size 3×3 and 256 feature maps.
- 4th convolutional layer with filter size 3×3 and 384 feature maps.
- 5th convolutional layer with filter size 3×3 and 256 feature maps.
- Fully connected layer with 1000 hidden units.
- Fully connected layer with seven hidden units.
- Softmax loss.

We have applied RELU layer (**activation layer**) to all convolutional and fully connected layers. Its aim is to fasten the convergence learning and to introduce the non-linearity to the proposed system. The RELU layer changes all the negative activation values to zero of the given input by applying the following function: f(x) = max (0, x). Max pooling layer (**down-sampling layer**) is applied after every RELU layer to reduce the spatial dimension with a filter of 3×3 and a stride of the length equal to two. The outputs of convolutional layer one and two after passing through activation layer are normalized using local response normalization. The networks weights are initiated using the **Xavier** [47] by drawing them from a uniform distribution on the interval [−scale, +scale].

Where scale = $\sqrt{\frac{3}{Fan_{in}}}$, Fan_{in} is the number of the input nodes [32]. The right weights value allows the convergence of the network in a reasonable time.

The following parameters have been used in learning phase:

- The standard gradient descend is used in training.
- Learning rate is set to 0.01.
- Dropout with 50%.

6. Handcrafted features classification

We have made a comparison between the features-based approach and CNN. The features-based classification method is composed of a feature extraction phase and a classification phase. Two features-based approaches are proposed, the first approach focuses on extracting six Mel Frequency Cepstrum Coefficients (MFCCs) statistics from the MFCC coefficients, while the second one uses the extracted Local Binary Patterns (LBP) features from the spectrograms.

6.1. MFCCs statistics-based classification

In the feature extraction phase, six MFCCs statistics, including mean, standard deviation, min, max, mean of the absolute difference, and standard deviation of absolute difference are calculated from the 12-MFCC coefficients and then stored in one vector. We have also tested the input normalization using zero means and the variance. The features are passed to training using three types of classifiers: SVM, K-nearest neighbor, and GMM.

MFCC is the most commonly used feature extraction method in automatic speech recognition [48]. We have compared two sets of features, the first features set only includes the 12 MFCC features while the second features include the 12 MFCC features, log energy, acceleration, and velocity. The 12 MFCC features give the best performance results with the following settings:

- Pre-emphasising Coefficient = 0.95
- Overlap = 50%
- Cepstrum Dimension = 12
- Triangular band-pass filters = 25
- The length of the discrete cosine transform DCT output = 12

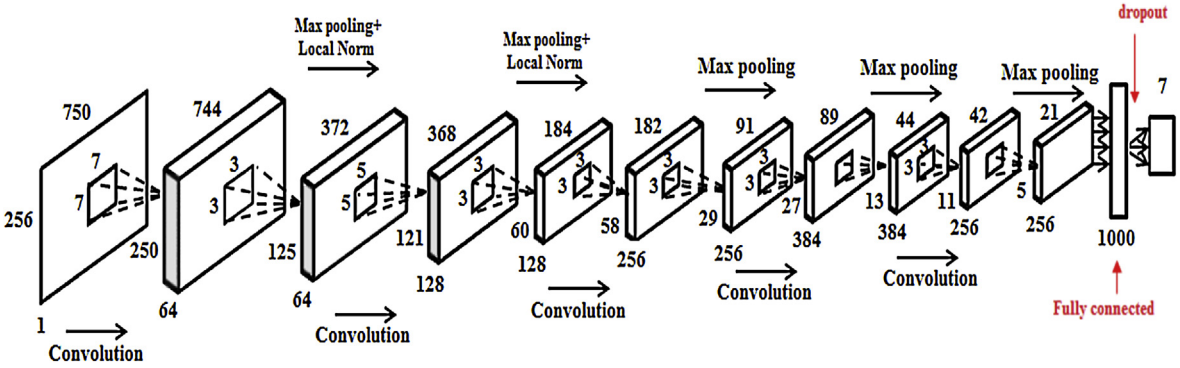

Fig. 1. The topology of the proposed convolutional neural network.

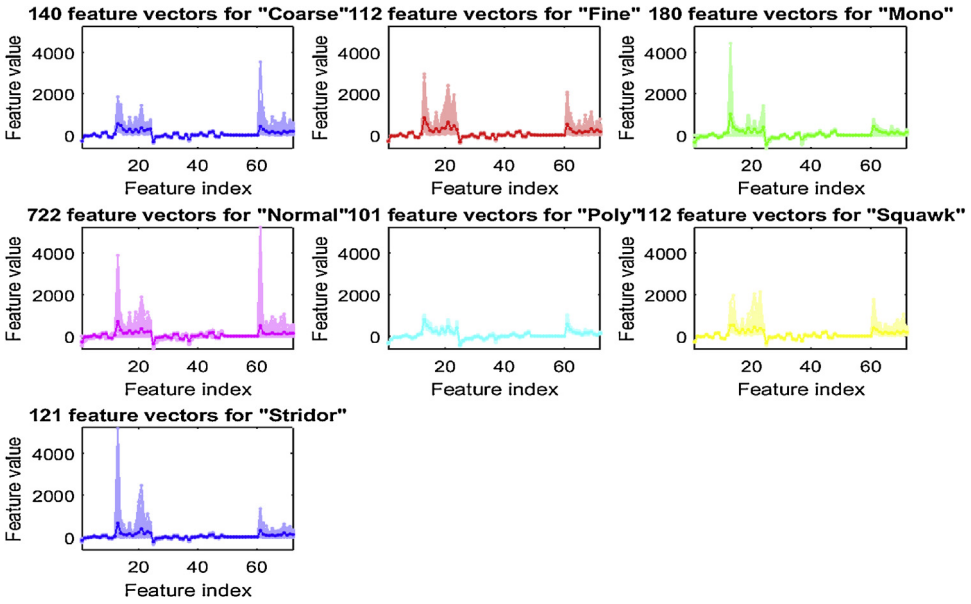

Fig. 2. The extracted features values within each class.

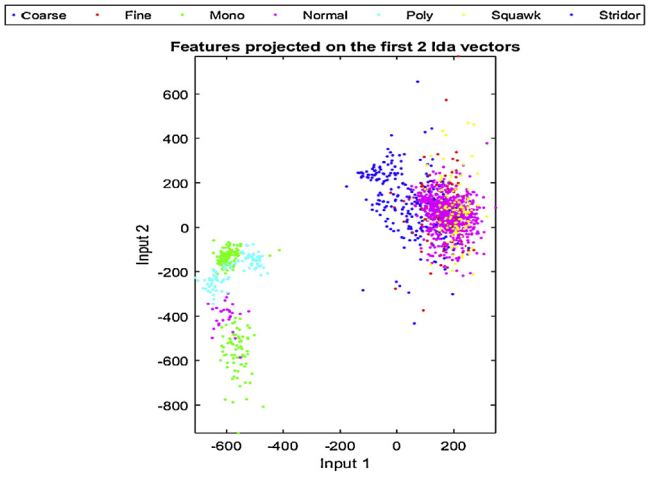

Fig. 3. The extracted features distribution using 2D projection.

The six statistics calculated from the 12-MFCC coefficients output a 72-dimensional feature vector. Fig. 2 shows the extracted features values within each class. Fig. 3 shows the distribution of the extracted features and where they are located through using 2D projection.

6.2. LBP features-based classification

We considered the spectrograms as texture, and we have extracted LBPs after converting each audio signal to a spectrogram having a size of 750×256. LBP is a straightforward and efficient texture operator which labels the pixels of an image by thresholding the neighborhood of every pixel and considers the outcome as a binary number [49]. It has been tested in various applications such as sound event classification [50], music genre classification [51–53], and environmental sound classification [54]. Dimensionality reduction and input whitening have been experimentally tested to enhance the performance. The aim of input whitening is to make the data less redundant. We have divided each image into 15 blocks and then using the basic LBP [55], a local histogram is built from each block. Finally, all the local histograms are aggregated to create a non-uniform global histogram. We have used 3×3-pixel neighborhoods which were found to be optimal in several applications. As the neighborhood consists of 8 pixels, each block gives a total of $2^8 = 256$ different labels. The aggregated histogram outputs a 3840-dimensional feature vector. Since its dimensionality is too high, we used principal component analysis to reduce the dimensionality.

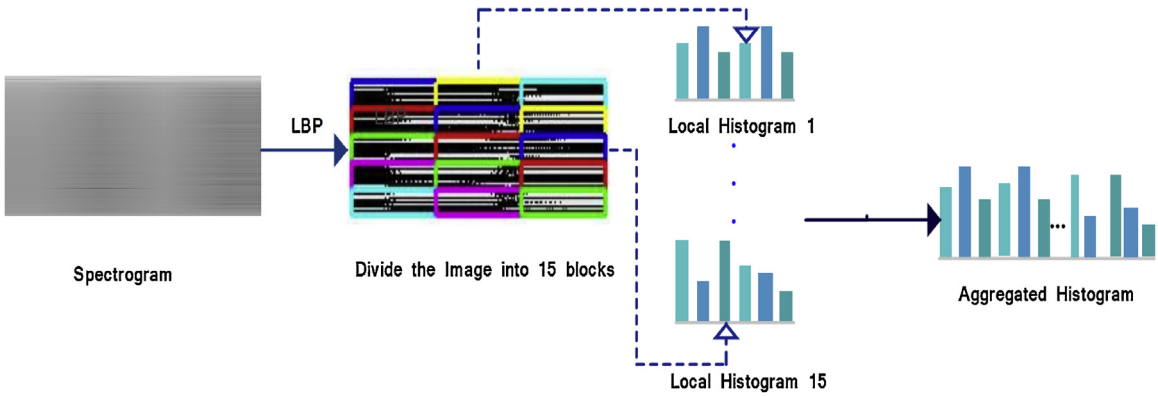

Fig. 4. Extracting LBP features process.

We have only retained features where the cumulative variance percentage is larger than a threshold having a value of 5%. Fig. 4 shows the steps of extracting LBP features.

6.3. The classifiers

The extracted features from MFCCs statistics and LBP were classified using three different classification methods: SVM, K-nearest neighbor, and GMM.

- SVM: The support vector machine classifier is a promising method for the classification of both linear and non-linear. SVM can deal with classification problems in many fields such as in speech recognition, text classification, and object classification [56–58]. In this work, we have used the LIBSVM [59] library to train the data.
- K-nearest neighbor: The k-nearest neighbor classifier is a straightforward and powerful predictive technique that can give competitive results. It was introduced in 1957 by Fix and Hodges [60]. The K-nearest neighbor classifier finds the closest k-nearest objects among the training data that are closest to the unknown input object [61]. The number of neighbors used here is 7 and to calculate the distance, we used Euclidean distance.
- GMM: Gaussian mixture models are probabilistic models which have been widely used in many classification problems. Using GMM, each class can be modelled by a Gaussian mixture model. During training, and for each class, the parameters are estimated through maximizing the likelihood. The parameters of the Gaussian distribution consist of the mean and the covariance matrix. The parameters for GMM are optimized using an iterative method, which is called expectation maximization (EM), and it chooses optimal parameters; K-means is used to get the initial iterations values. There are three types of the covariance matrix (spherical, diagonal, and full covariance matrix). In this work, we have used a diagonal covariance matrix due to its computation efficiency and preciseness.

7. Results

The dataset has been randomly split into 70% training and 30% testing. Twenty-five percent of the training data is retained and used for cross-validation to select the best models parameters. After selecting the best parameters, it has been added to the training data. The same testing dataset is used in all the experiments. Input normalization has also been tested through applying zero mean and unity variance to the calculated MFCCs statistics and applying whitening to the features extracted from LBP. The dataset augmentation techniques are only applied to the CNN approach where each spectrogram is randomly cropped and perturbed using a VTLN technique. The results in Section 7.1 are related to the CNN classification performance applied to the original dataset and the dataset with augmentation, as well as the results of ensembling many models. The results of each test are given in term of accuracy (% of correctly classified instances). The learning rate curves for testing and training loss are given below for each topology. We have also tested 'CNN+SVM' configuration through replacing the fully-connected layer with linear support vector machines and 'Features+CNN' configuration as well, given in Sections 7.1.4 and 7.1.5 respectively. In Section 7.2, we made a comparison between the performances of the handcrafted features-based approaches with CNN.

Table 3
The performance of CNN applied to the original dataset with different batch sizes and iterations numbers.

Iterations/Batch Size	32	64	128
5000	86.98	85.94	90.89
10,000	89.59	92.19	92.71
15,000	92.45	91.92	92.71
20,000	92.34	92.19	**93.26**

7.1. CNN results

7.1.1. Baseline configurations with dataset augmentation

Following the proposed CNN topology in Section 5, the original set of spectrograms extracted from the audio files were trained for 20,000 iterations, with different batch sizes (1, 16, 32, 64 and 128). The batch size heavily contributes towards determining learning parameters, which affect the accuracy of the model. Small batch sizes led to a less accurate estimation of the gradient, when working with images. The performance accuracy of the model using batch size equal to 1 and 16 was less than 10%. The best obtained accuracy was **93.26%,** with a batch size equal to **128,** which trained for 20,000 iterations. The results of different batch sizes with different iteration numbers are given in Table 3. The training log and test accuracy curves, with respect to the number of iterations, are shown in Fig. 5(a). After augmenting the training dataset by applying random cropping and VTLC, and by training the model for 20,000 iterations, we reached an accuracy of **95.10%.** The corresponding training log and test accuracy curves are given in Fig. 5(b). We have also provided the accuracies, taken at different iterations with different batch sizes, and applied to the augmented data, in Table 4. The convergence times of the CNN models applied to the original and the augmented data sets are provided in Table 5. The first model took 38 min and 35 s while the second CNN model took 3 h, and 28 min and 39 s. Both models were trained on Tesla K40 GPU.

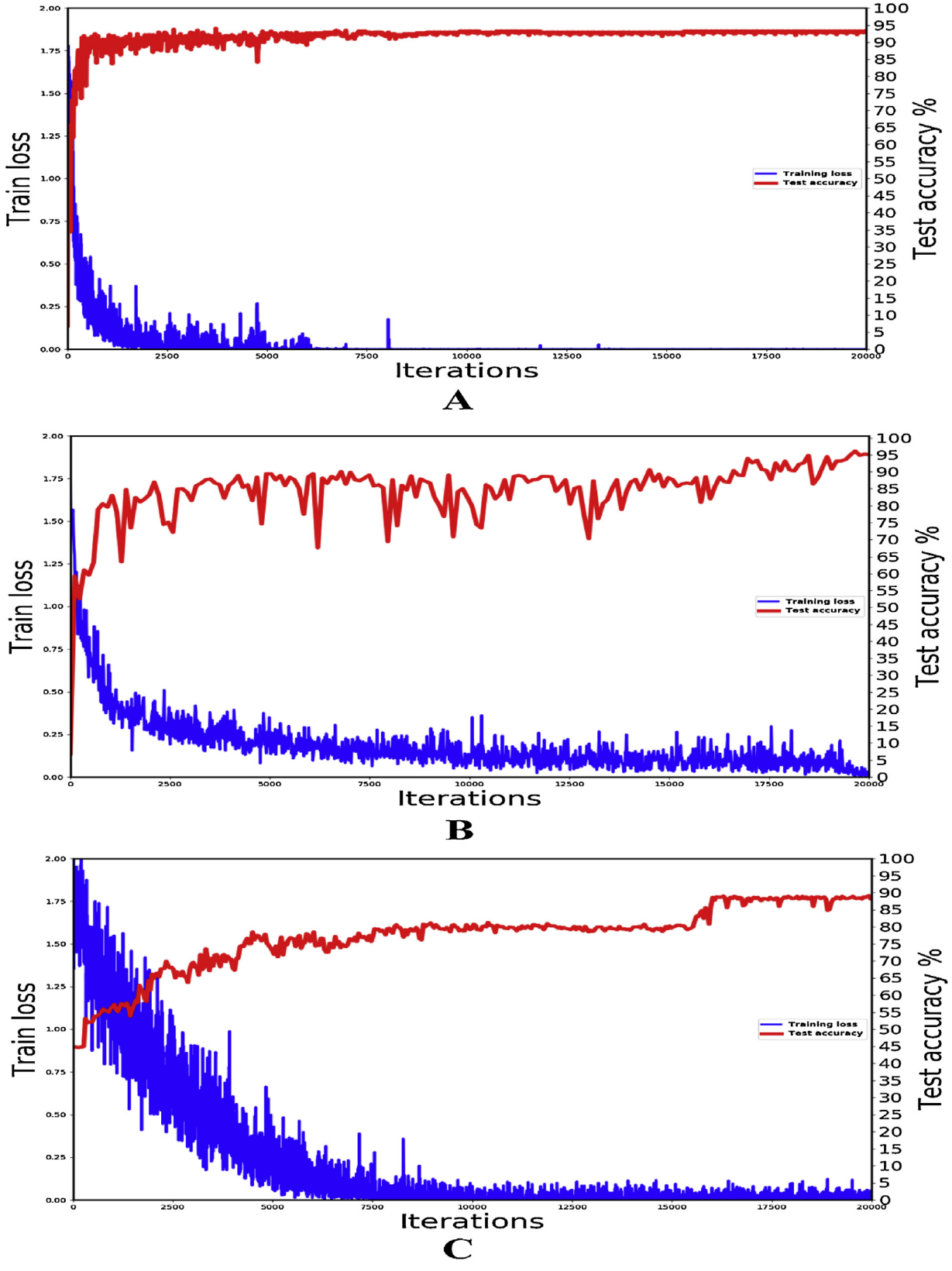

Fig. 5. CNN results: Log loss and test accuracy of the tested dataset: (a). Accuracy results of the original baseline dataset (batch size = 128), (b). Accuracy results of the augmented dataset with ten times random cropping and vocal tract length perturbation, (c). Accuracy results of the CNN + Classifier configuration.

7.1.2. Ensemble model

Since it is known that an ensemble is usually more accurate than a single model, we thought of using an ensemble to enhance the accuracy. Convolutional neural networks have a final fully connected layer with Softmax activation, which can provide a vector of probabilities. The probabilities correspond to an input image belonging to a particular class. In this work, we opted to sum up the probabilities of the models that gave the highest accuracies, taken

Table 4
The performance of CNN applied to the augmented dataset with different batch sizes and iterations numbers.

Iterations/Batch Size	32	64	128
5000	81.62	78.10	89.59
10,000	88.82	85.15	79.63
15,000	91.73	88.82	87.60
20,000	94.18	93.42	**95.10**

Table 5
Convergence time of CNN applied to the original and the augmented data sets.

	Original Data set	Augmented Data set
Training time	38 min and 35 s	3 h, 28 min and 39 s

at different iterations, with a batch size equal to128. The Softmax activation outputs of the following four models are used:

- The output of the CNN model applied to augmented data taken after 20,000 iterations.
- The output of the CNN model applied to augmented data taken after 19,656 iterations.
- The output of the CNN model applied to original data taken after 2286 iterations.
- The output of the CNN model applied to original data taken after 2664 iterations

By summing up the Softmax activations output, we achieved an accuracy of **95.56%.**

7.1.3. Use of linear SVM on the top of CNN

We decided to investigate a hybrid approach 'CNN+ Classifier'. Following the same used in [32], we have put downstream of the last convolutional layer a small fully-connected layer, with a number of neurons equal to seven, which is the number of classes, and without using activation layer (non-linearities), feeding a hinge loss layer. As mentioned in [32], learning this architecture is equivalent to work with a linear SVM acting on features learned by the CNN. The accuracy obtained was 88.54% which is less than the accuracy reached by using CNN in Section 7.1.1. The decreasing result is due to the linearity of support vector machines Fig. 5(c).

7.1.4. Use of CNN instead of traditional classifiers to classify MFCCs statistics

In order to test an alternative 'Features + CNN' configuration, and instead of using traditional classifiers such as SVM, K-nearest neighbors, and GMM to classify MFCCs statistics extracted from the signals, we replaced the traditional classifiers by fully-connected layers, and we gave the set of features as input to CNN. The used CNN consists of two fully-connected layers. The first fully-connected layer has 2400 outputs followed by a RELU function with a dropout of 50%. The second fully-connected layer has seven outputs which are the number of classes without dropout or RELU function. The network weights are initiated using Xavier. The regularizing parameter weight decay is set to 0.001. The number of iterations is set to 1 million iterations. The learning rate is initiated to 0.0001, and the used policy is inverse decay. We have tested many batch size values (1, 16, 32, 64 and 128). The standard gradient descend is used in training. This topology gave an accuracy of **91.67%,** after 1 million iterations, when the batch size used was equal to one. The same topology gave an accuracy of **100%** at the following iterations: 160,000, 240,000, 700,000, 780,000 and 940,000. The training log and test accuracy curves, with respect to the number of iterations, are shown in Fig. 6(a). The accuracies of different batch sizes taken at different iterations are given in Table 6.

Table 6
The accuracy performance of the MFCC's statistics features trained with CNN with different batch sizes and iterations numbers.

Iterations/Batch size	1	16	32	64	128
100,000	75	86.46	86.72	86.59	86.65
160,000	**100**	85.94	86.20	87.37	86.91
240,000	**100**	87.5	87.5	87.63	86.98
700,000	**100**	83.33	85.68	87.5	86.98
780,000	**100**	88.54	85.94	87.76	86.65
940,000	**100**	83.85	86.46	87.11	86.39
1million	**91.67**	84.38	85.42	87.5	86.65

Table 7
The accuracy performance of the LBP features trained with CNN with different batch sizes and iterations numbers.

Iterations/Batch size	1	16	32	64	128
120,000	40	63.75	69.38	64.06	64.53
240,000	60	72.5	61.25	62.5	64.38
510,000	**100**	58.75	61.88	63.75	65.63
600,000	**100**	70	61.88	64.69	66.09
630,000	**100**	67.5	71.88	64.38	63.44
1million	**80**	67.5	66.88	63.44	66.56

Table 8
The accuracy performance of the MFCC's statistics classified with SVM, K-nearest neighbor, and GMM.

	Input Normalization	SVM	KNN	GMM
Accuracy	1	**91.12**	79.94	86.68
	0	88.97	72.89	82.85

7.1.5. Use of CNN instead of traditional classifiers to classify LBP features extracted from the set of spectrograms

Similar to Section 7.1.4, and instead of using traditional classifiers to classify the set of local binary pattern features extracted from the spectrograms, we opted for a CNN topology which takes as input the LBP features. The CNN consists of three fully-connected layers. The first layer has 1000 outputs followed by an activation layer (RELU function) with a dropout of 50%. The second layer has 50 outputs followed by the third fully-connected layer with seven outputs. The parameters values are set as in Section 7.1.4, and the regularizing parameter weight decay value is set to 0.5. This configuration allowed us to reach an accuracy of **80%** after 1 million iterations, when the batch size used was equal to one. The training log and test accuracy curves with respect to the number of iterations are shown in Fig. 6(b). The accuracies of different batch sizes taken at different iterations are given in Table 7. The same topology reached an accuracy of **100%** at three different iterations as shown in Table 7.

7.2. Performance comparison

The obtained results provided, consider the best configuration parameters that offer the best accuracy. For the first feature-based approach, where six MFCCs statistics were extracted from 12-MFCC coefficients, we reached an accuracy of **91.12%** using support vector machines. The configuration 'Features + CNN' allowed us to reach an accuracy of **100%** at many iterations (160,000, 240,000, 700,000, 780,000 and 940,000), and **91.67%** after 1 million iterations. This configuration outperformed the traditional 'Features + Classifier' approach. In comparison with SVM, K-NN, and GMM, SVM outperformed the two classifiers, where RBF (Radial Basis Function) kernel is used to train the features. Input normalization had an influence on the final results, where the accuracy of SVM increased from **88.97%** to **91.12%**. The performance of the classifiers, with and without input normalization, is given in Table 8. The features-

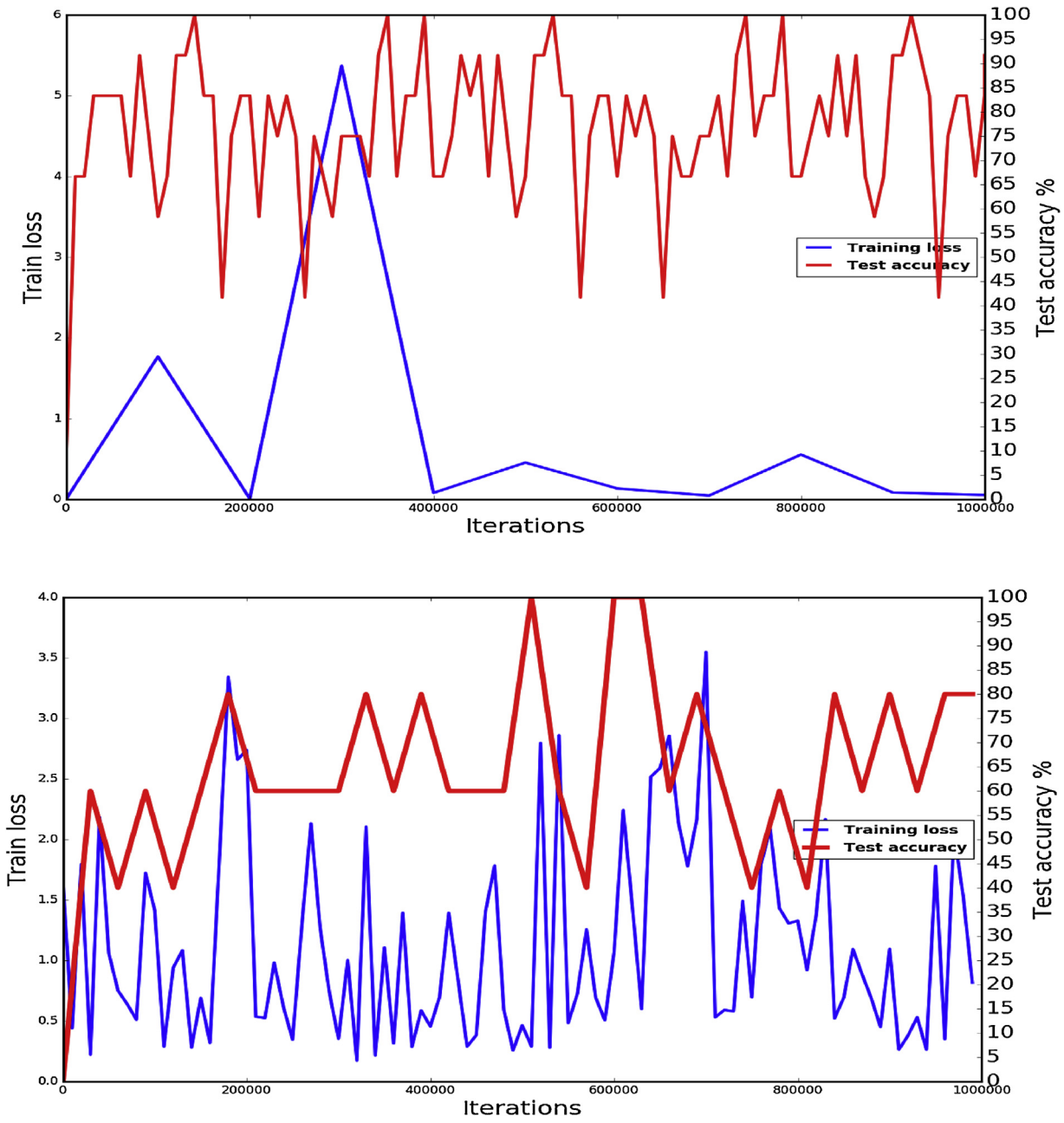

Fig. 6. Features + CNN configuration results: Log loss and test accuracy of the tested dataset: (a). Accuracy results of the extracted MFCC's statistics classified with CNN instead of the traditional classifiers, (b). Accuracy results of the extracted LBP classified with CNN.

based approaches are implemented using a speech audio signal processing toolbox [62] and machine learning toolbox [63].

For the second approach, where LBPs features are extracted from the spectrograms, we reached an accuracy of **71.21%** using support vector machines, while the configuration 'Features+CNN' allowed us to reach an accuracy of **100%** at the following iterations: 510,000, 600,000 and 630000, and **80%** after 1 million iterations. The performance of the classifiers, with and without input normalization, is given in Table 9. The poor performance of configuration "LBP+SVM" is due to the very low variance of intensities within the image. Most instances of fine crackle are misclassified as coarse crackle, due to the similarities between the two classes, when using this configuration.

The results related to CNN are given in Sections 7.1.1 and 7.1.2. Using the original data, we reached an accuracy level of **93.26%,** and by applying augmentation techniques, this increased to **95.10%**.

Table 9
The accuracy performance of the LBPs features classified with SVM, K-nearest neighbor, and GMM.

	Whitening	SVM	KNN	GMM
Accuracy	1	**71.21**	67.28	69.07
	0	70.59	66.51	68.91

Our model ensemble showed its efficiency in improving the performance of CNN, where the accuracy reached **95.56%**. Table 10 provides the performances of the different configurations. The accuracy of "MFCC-CNN" and "LBP-CNN" are given after 1 million iterations. Confusion matrixes relating to CNN ensembling, MFCC-SVM, and LBP-SVM are given in Fig. 7 (a)–(c) respectively. We have also calculated precision-recall values for each class, which is provided in Table 11.

Table 10
Performance comparison between different configurations.

	Ensembling CNN	MFCC-SVM	LBP-SVM	MFCC-CNN	LBP-CNN
Accuracy	**95.56**	91.12	71.21	91.67	80.00

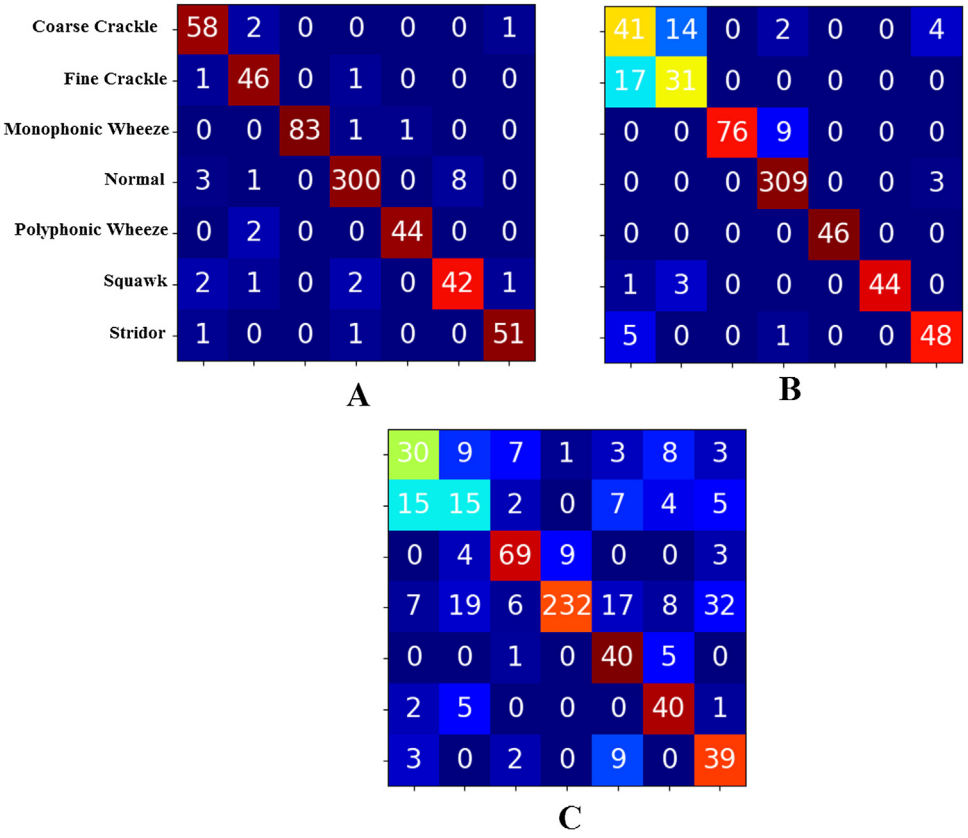

Fig. 7. Confusion matrixes: (a). Confusion matrix for the ensembling of three CNN models applied to the augmented dataset, (b). Confusion matrix for the 12-MFCC's statistics classified with SVM, (c). Confusion matrix for the LBPs features classified with SVM.

Table 11
Precision and recall comparison between CNN and features-based approaches.

Class	CNN		MFCC-SVM		LBP-SVM	
	Precision	Recall	Precision	Recall	Precision	Recall
Coarse Crackle	0.95082	0.89	0.67	0.64	0.49	0.53
Fine Crackle	0.96	0.88	0.64	0.64	0.31	0.29
Monophonic Wheeze	0.98	1	0.89	1.00	0.81	0.79
Normal	0.96	0.98	0.99	0.96	0.74	0.96
Polyphonic Wheeze	0.96	0.98	1.00	1.00	0.87	0.53
Squawk	0.88	0.84	0.92	1.00	0.83	0.61
Stridor	0.96	0.96	0.90	0.88	0.73	0.53

8. Discussion

The auscultation is the most common technique used by physicians to evaluate lungs, becoming a routine part of the clinical examination. However, this technique is a subjective process [64], and also lung sounds are non-stationary, complicating the tasks of analysis, recognition, and distinction. Therefore, with the emergence of machine learning and artificial intelligence, there was an interest in automating this process and developing automatic recognition systems that can help to overcome these limitations. Many studies have been conducted in order to analyze and classify lung sounds. Some studies focused on the binary classification of lung sounds into crackle and non-crackle [14], normal respiratory sounds and continuous adventitious sounds [15], normal and wheezing sounds [16], crackle and normal [21]. Some focused on more than two classes such as [4], where the lung sounds were classified into normal, crackles, and rhonchus. In [5], the lung sounds were classified into five classes including normal sounds, coarse crackle, fine crackle, monophonic wheeze, and polyphonic wheeze. These studies have used handcrafted-based approaches where firstly a set of features are extracted and then are fed into a classifier for training in order to build a predictive model. Each study has used different type of feature, such as MFCC [16], higher-order statistics [5], time-frequency analysis [14], temporal features [21] and others [4,15] along with different classifiers, such as support vector machines [4,14,15], Gaussian mixture models [16,21], K-nearest neighbor separately [14] and combined with Bayes classifier [5] and multi-layer perceptron [14]. The features extraction step allows the

extracting of a set of discriminant and non-redundant data from the input data for each class. These features are then used to find patterns and build a predictive model using a classifier. However, the problem with this approach is the variety of the features and the classifiers, which leads one to ask: "Which features and classifiers should be used?" "Which ones will work better?" The solution for this problem came with the emergence of deep learning, where a set of self-adaptive features can be extracted, corresponding to a specific task. In addition, these features are regulated cognitively by a countless number of parameters relating to the mechanism of the human brain, which will provide more precise and efficient results [65]. In this work, we have assessed the performance of both handcrafted features-based and deep learning-based approaches for classifying lung sounds into seven classes. In the handcrafted features approach, we used two types of engineered features. We relied on signal processing in order to extract MFCC's features from the signals; and also on image processing to extract texture features (LBP) from the spectrograms. For the deep learning, we used CNN, which is very suitable for image classification (spectrograms in our case) and it has filters which act like features detectors. The use of CNN requires the design of the right topology through defining the number of layers and the regularization parameters. From the achieved results, we found that CNN outperformed the handcrafted features-based approach and provided promising results. Convolutional neural networks are now state-of-the-art and they can deal well with any kind of classification tasks in the medical fields, with their performance depending on various factors such as the chosen topology, the size the training samples, and parameters/hyperparameters selection.

9. Conclusion

In this work, we used convolutional neural networks to classify lung sounds through the use of three types of inputs: Spectrograms, MFCC features and LBP features. CNNs have become state-of-the-art, and they can solve challenging classification tasks. However, their performance depends on learning parameters, batch size and of iterations. Convolutional neural networks can replace traditional classfiers through the use of fully-connected layers to train the features (MFCC and LBP) which helped increase performance, as shown in Sections 7.1.4 and 7.1.5. Among the methods used to improve the performance of CNN is ensembling through summing up the output of Softmax activation of four CNN models. By using this method, we successfully increased the accuracy from 95.10% to 95.56%. In our upcoming work, we will opt for testing larger CNNs topologies as well as using recurrent neural network and their combinations with convolutional neural networks.

Lung sound processing methods are useful if they are straightforward and can accurately recognize more lung sounds classes. The aim of our proposed work and the previous work related to lung sound classification is to facilitate clinical decision making. The proposed automatic pattern recognition-based system can be embedded with an electronic stethoscope to deal with the limitations of the traditional auscultation technique.

Acknowledgments

This work was supported by the China Scholarship Council (CSC). The authors would also like to thank Mr. Chris Carson for providing the material for the creation of the dataset.

References

[1] Moussavi Zahra. Fundamentals of respiratory sounds and analysis. Synth Lect Biomed Eng 2006;1(1):1–68.
[2] Sengupta Nandini, Sahidullah Md, Saha Goutam. Lung sound classification using cepstral-based statistical features. Comput Biol Med 2016;75:118–29.
[3] Breath Sounds - Symptoms, Causes, Tests – NY Times Health Information. Nytimes.com. N.P., 2017. Web. 27 Feb. 2017.
[4] İçer Semra, Gengeç Şerife. Classification and analysis of non-stationary characteristics of crackle and rhonchus lung adventitious sounds. Digit Signal Process 2014;28:18–27.
[5] Naves Raphael, Barbosa Bruno HG, Ferreira Danton D. Classification of lung sounds using higher-order statistics: a divide-and-conquer approach. Comput Methods Prog Biomed 2016;129:12–20.
[6] Chang Gwo-Ching, Lai Yung-Fa. Performance evaluation and enhancement of lung sound recognition system in two real noisy environments. Comput Methods Prog Biomed 2010;97(2):141–50.
[7] Chang Gwo-Ching, Cheng Yi-Ping. Investigation of noise effect on lung sound recognition, 2008. In: International Conference on. Vol. 3. IEEE. 2008.
[8] Reichert Sandra, et al. Analysis of respiratory sounds: state of the art. Clin Med Insights Circ Respir Pulm Med 2008;2:45.
[9] American Thoracic Society. Updated nomenclature for membership relation. ATS News 3; 1977. p. 5–6.
[10] Sovijärvi ARA, Vanderschoot J, Earis JE, editors. Computerized respiratory sound analysis (CORSA): recommended standards for terms and techniques: ERS task force report. Munksgaard; 2000.
[11] Dokur Zümray. Respiratory sound classification by using an incremental supervised neural network. Pattern Anal Appl 2009;12(4):309.
[12] Numanoğlu N. Respiratory system and diseases. Ankara: Antıplnc; 1997. p. 71–5.
[13] Lehrer S. Understanding lung sounds. 3rd ed. Philadelphia: WB Saunders; 2002.
[14] Serbes Gorkhem, Okan Sakar C, Kahya Yasemin P, Aydin Nizamettin. Pulmonary crackle detection using time–frequency and time–scale analysis. Digit Signal Process 2013;23(3):1012–21.
[15] Jin Feng, Sattar Farook, Goh Daniel YT. New approaches for spectro-temporal feature extraction with applications to respiratory sound classification. Neurocomputing 2014;123:362–71.
[16] Bahoura Mohammed. Pattern recognition methods applied to respiratory sounds classification into normal and wheeze classes. Comput Biol Med 2009;39(9):824–43.
[17] Reyes BA, Charleston-Villalobos Sonia, González-Camarena Ramón, Aljama-Corrales Tomás. Assessment of time–frequency representation techniques for thoracic sounds analysis. Comput Methods Prog Biomed 2014;114(3):276–90.
[18] Orjuela-Cañón Alvaro D, Gómez-Cajas Diego F, Jiménez-Moreno Robinson. Artificial neural networks for acoustic lung signals classification. In: Iberoamerican congress on pattern recognition. Springer International Publishing; 2014.
[19] Mayorga P, Druzgalski C, Morelos RL, González OH, Vidales J. Acoustics based assessment of respiratory diseases using GMM classification. In: Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE. 2010.
[20] Mayorga P, Druzgalski C, González OH, Zazueta A, Criollo MA. Expanded quantitative models for assessment of respiratory diseases and monitoring. In: Health Care Exchanges (PAHCE), 2011 Pan American. 2011.
[21] Maruf SO, Azhar MU, Khawaja SG, Akram MU. Crackle separation and classification from normal respiratory sounds using gaussian mixture model. In: Industrial and Information Systems (ICIIS), 2015 IEEE 10th International Conference on. 2015.
[22] Khodabakhshi Mohammad Bagher, Moradi Mohammad Hassan. The attractor recurrent neural network based on fuzzy functions: an effective model for the classification of lung abnormalities. Comput Biol Med 2017;84:124–36.
[23] Mastorocostas Paris, Stavrakoudis Dimitris, Theocharis John. A pipelined recurrent fuzzy model for real-time analysis of lung sounds. Eng Appl Artif Intell 2008;21(8):1301–8.
[24] Mastorocostas PA, Tolias YA, Theocharis JB, Hadjileontiadis LJ, Panas SM. An orthogonal least squares-based fuzzy filter for real-time analysis of lung sounds. IEEE Trans Biomed Eng 2000;47(9):1165–76.
[25] Yeginer Mete, Kahya Yasemin P. Elimination of vesicular sounds from pulmonary crackle waveforms. Comput Methods Prog Biomed 2008;89(1):1–13.
[26] Pinho Cátia, Oliveira Ana, Jácome Cristina, Rodrigues João, Marques Alda. Automatic crackle detection algorithm based on fractal dimension and box filtering. Procedia Comput Sci 2015;64:705–12.
[27] Abdel-Hamid Ossama, Abdel-rahman Mohamed, Jiang Hui, Deng Li, Penn Gerald, Yu Dong. Convolutional neural networks for speech recognition. IEEE/ACM Trans Audio Speech Lang Process 2014;22(10):1533–45.
[28] LeCun Yann, Bottou Léon, Bengio Yoshua, Haffner Patrick. Gradient-based learning applied to document recognition. Proce IEEE 1998;86(11):2278–324.
[29] Krizhevsky Alex, Sutskever Ilya, Hinton Geoffrey E. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012.
[30] Szegedy Christian, Liu Wei, Jia Yangqing, Sermanet Pierre, Reed Scott, Anguelov Dragomir, et al. Going deeper with convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015.
[31] Girshick Ross, Donahue Jeff, Darrell Trevor, Malik Jitendra. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2014.

[32] Ferrari Alessandro, Lombardi Stefano, Signoroni Alberto. Bacterial colony counting with convolutional neural networks in digital microbiology imaging. Pattern Recognit 2017;61:629–40.
[33] Salamon Justin, Pablo Bello Juan. Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Process Lett 2017;24(3):279–83.
[34] Gajhede Nicolai, Beck Oliver, Purwins Hendrik. Convolutional neural networks with batch normalization for classifying Hi-hat, Snare, and bass percussion sound samples. In: Proceedings of the Audio Mostly 2016. 2016.
[35] Amoh Justice, Odame Kofi. DeepCough: a deep convolutional neural network in a wearable cough detection system. In: Biomedical Circuits and Systems Conference (BioCAS), 2015 IEEE. 2015.
[36] Nilanon Tanachat, Yao Jiayu, Hao Junheng, Purushotham Sanjay, Liu Yan. Normal/abnormal heart sound recordings classification using convolutional neural network. In: Computing in Cardiology Conference (CINC), 2016. 2016.
[37] Potes Cristhian, Parvaneh Saman, Rahman Asif, Conroy Bryan. Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds; 2016.
[38] The R.A.L.E. Repository. Rale.ca. N.P., 2017. Web. 28 Feb. 2017.
[39] Sovijarvi ARA, Vanderschoot J, Earis JE. Standardization of computerized respiratory sound analysis – CORSA. Eur Respir Rev 2000;10:585.
[40] Palaniappan R, Sundaraj K, Lam CK. Reliable system for respiratory pathology classification from breath sound signals. In: System Reliability and Science (ICSRS), International Conference on. IEEE; 2016.
[41] Lee Li, Rose Richard. A frequency warping approach to speaker normalization. IEEE Trans Speech Audio Process 1998;6(1):49–60.
[42] Jaitly Navdeep, Hinton Geoffrey E. Vocal tract length perturbation (VTLP) improves speech recognition. In: Proc. ICML Workshop on Deep Learning for Audio, Speech and Language; 2013.
[43] Ko Tom, Peddinti Vijayaditya, Povey Daniel, Khudanpur Sanjeev. Audio augmentation for speech recognition. INTERSPEECH; 2015.
[44] Cui Xiaodong, Goel Vaibhava, Kingsbury Brian. Data augmentation for deep neural network acoustic modeling. IEEE/ACM Trans Audio Speech Lang Process (TASLP) 2015;23(9):1469–77.
[45] Jia Yangqing, Shelhamer Evan, Donahue Jeff, Karayev Sergey, Long Jonathan, Girshick Ross, et al. Caffe: convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM International Conference on Multimedia. 2014.
[46] Krizhevsky Alex, Sutskever Ilya, Hinton Geoffrey E. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012.
[47] Glorot Xavier, Bengio Yoshua. Understanding the difficulty of training deep feedforward neural networks, vol. 9. Aistats; 2010.
[48] O'Shaughnessy Douglas. Invited paper: automatic speech recognition: history, methods and challenges. Pattern Recognit 2008;41(10):2965–79.
[49] Pietikäinen Matti. Local binary patterns. Scholarpedia 2010;5(3):9775.
[50] Ren Jianfeng, Jiang Xudong, Yuan Junsong, Magnenat-Thalman Nadia. Sound-event classification using robust texture features for robot hearing. IEEE Trans Multimedia 2017;19(3):447–58.
[51] Costa Yandre MG, Oliveira LS, Koerich Alessandro L, Gouyon Fabien, Martins JG. Music genre classification using LBP textural features. Signal Process 2012;92(11):2723–37.
[52] Wu Hai Qian, Zhang Ming. Gabor-lbp features and combined classifiers for music genre classification. Advanced materials research, vol. 756. Trans Tech Publications; 2013.
[53] Agera Nelson, Chapaneri Santosh, Jayaswal Deepak. Exploring textural features for automatic music genre classification. In: Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on. 2015.
[54] Kobayashi Takumi, Ye Jiaxing. Acoustic feature extraction by statistics based local binary pattern for environmental sound classification. In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. 2014.
[55] Ojala Timo, Pietikäinen Matti, Harwood David. A comparative study of texture measures with classification based on featured distributions. Pattern Recognit 1996;29(1):51–9.
[56] Cortes Corinna, Vapnik Vladimir. Support-vector networks. Mach Learn 1995;20(3):273–97.
[57] Chapelle Olivier, Haffner Patrick, Vapnik Vladimir N. Support vector machines for histogram-based image classification. IEEE Trans Neural Netw 1999;10(5):1055–64.
[58] Tsai Chun-Wei, Cho Keng-Mao, Yang Wei-Shan, Su Yi-Ching, Yang Chu-Sing, Chiang Ming-Chao. A support vector machine based dynamic classifier for face recognition. Int J Innovative Comput Inf Control 2011;7(6):3437–55.
[59] Chang Chih-Chung, Lin Chih-Jen. LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol (TIST) 2011;2(3):27.
[60] Fix Evelyn, Hodges Jr Joseph L. Discriminatory analysis-nonparametric discrimination: consistency properties. California: Univ Berkeley; 1951.
[61] Garcia Vincent, Debreuve Eric, Nielsen Frank, Barlaud Michel. K-nearest neighbor search: fast GPU-based implementations and application to high-dimensional feature matching. In: Image Processing (ICIP), 2010 17th IEEE International Conference on. 2010.
[62] Jyh-Shing Roger Jang Speech and Audio Processing (SAP) Toolbox. Available at ¨http://mirlab.org/jang/matlab/toolbox/sap..
[63] Jyh-Shing Roger Jang. Machine Learning Toolbox. Available at http://mirlab.org/jang/matlab/toolbox/machineLearning.
[64] Andrès Emmanuel, Reichert Sandra, Gass Raymond, Brandt Christian. A French national research project to the creation of an auscultation's school: the ASAP project. Eur J Intern Med 2009;20(3):323–7.
[65] Min Seonwoo, Lee Byunghan, Yoon Sungroh. Deep learning in bioinformatics. Brief Bioinform 2017;18(5):851–69.