Third International Conference on "Materials, Computing and Communication Technologies"
in association with International Journal of Scientific Research in Science,
Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)

A CNN Model with a Parallel Pooling Structure Used to Classify Lung Sounds

R. Robert*[1], A. Nanthini[2], V. Kavitha[2], R. V. Rabisha[2], L. Liju[2]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Annai Vailankanni College of Engineering, Tamil Nadu, India

[2]B.E, Third-Year Students, Department of Electronics and Communication Engineering, Annai Vailankanni College of Engineering, Tamil Nadu, India

ABSTRACT

The early diagnosis of respiratory disorders relies heavily on the detection of distinct lung sounds recorded using electronic stethoscopes. Machine learning techniques have been heavily used in the last 30 years to improve the accuracy of specialist evaluations. In this paper, a new pre-trained Convolutional Neural Network (CNN) model for deep feature extraction is proposed. An average-pooling layer and a max-pooling layer are connected in parallel in the CNN architecture to improve classification performance.. Using the Random Subspace Ensembles (RSE) approach, the deep features are used as input to the Linear Discriminant Analysis (LDA) classifier. The proposed technique was tested using the ICBHI 2017 dataset, which is a difficult dataset. When compared to other existing approaches utilizing the same dataset, deep features and the LDA with the RSE method delivered the best accuracy score, improving classification accuracy by 5.75 percent.

INDEX TERMS: Lung sound, CNN model, parallel pooling, deep features, RSE method.

I. INTRODUCTION

Lung disease ranks third among fatality causes worldwide. According to the World Health Organization (WHO), more than 3 million people die each year due to respiratory diseases [1]. Lung sound attributes and their diagnosis play a significant role in pulmonary pathology.

Lung sounds can generally be grouped as "normal lung sounds" or "abnormal lung sounds." Normal lung sounds are when no pulmonary disease exists, whilst abnormal lung sounds are heard when a pulmonary disease is present [2], [3]. An abnormal lung sound is a supplementary respiratory sound that is heard in addition to the normal lung sound. Abnormal lung sounds are known as continuous if they contain wheezes, and discontinuous if they contain crackles. The presence of such sounds mostly indicates the presence of lung disease [4].

Auscultation is a method by which doctors evaluate and diagnose lung diseases using a stethoscope. It is known as a low-cost, easy-to-apply, and reliable test that requires minimal diagnosis duration [5]. The test is able to provide considerable information about lung diseases and their symptoms [6]; however, the classical

auscultation process using a stethoscope is not infallible as it depends on the skill of the physician and their hearing sensitivity.

Because of the inclusion of non-stationary signals, lung sounds can be difficult to analyze and separate using conventional auscultation techniques. Hence, the use of an electronic stethoscope combined with an artificial intelligence system can be used as a means to overcoming the limitations of conventional auscultaton, and thereby providing a more reliable and efficient method through automated diagnosis [7].

From the outset of machine learning and pattern recognition, numerous studies have put forwards proposed methods for the automatic classification of lung sounds. In the literature, conventional methods have generally been used, consisting of classifiers and hand-crafted features for the categorization of lung sounds. In [6], features are extracted with the frequency ratio of Power Spectral Density (PSD) values and the Hilbert-Huang Transform (HHT) method, and then evaluated using Support Vector Machine (SVM) algorithm. In [8], the features extracted from time- frequency and time-scale analysis methods are utilized for the detection of normal lung sounds and crackles, with k- Nearest Neighbors (k-NN), Multilayer Perceptron (MLP) and SVM used for the classification stage. The best accuracy was achieved with the SVM. In [9], the feature set is constituted by instantaneous kurtosis, discriminating function, and entropy in order to classify normal lung sounds and abnormal lung sounds, which consist of wheezes, stridor, and rhonchi. The best classification accuracy was achieved with the SVM classifier. In [10], the Mel-Frequency Cepstral Coefficients (MFCCs) was used in order to extract features from respiratory signals, and the proposed approach was evaluated using the Gaussian Mixture Model (GMM). In [7], Higher Order Statistics (HOS) were used for feature extraction, with genetic algorithms and Fisher's discriminant ratio applied to the feature set for the purposes of feature reduction. For classifying lung sounds, which include normal, coarse crackle, fine crackle, as well as monophonic and polyphonic wheezes, k-NN and Naive Bayes classifiers were employed. In [11], the feature set was constituted of autoregressive model coefficients, wavelet coefficients and some parameters of crackles, with k-NN and Artificial Neural Network (ANN) employed in the classification stage. Recently, deep learning-based models have been used for sound classification, since these models mostly provide a better level of performance over conventional methods [12]. In [13], CNNs were proposed for environmental sound classification using spectrogram images conveyed as input to the CNN. In [14], a CNN model was employed for the classification of lung sounds, with the CNN shown to perform superior to MFCC features in the SVM. In [15], three methods were proposed for the categorization of respiratory sounds. First, MFCC features were used in the GMM, SVM, and k-NN classifiers. In the second method, Local Binary Pattern (LBP) features were used in the GMM, SVM, and k-NN classifiers, whilst for the third method, the CNN model was used in both the training and testing stages.

Overall, the best classification accuracy was achieved using the CNN model of the third method.

In the method proposed in the current study, a hybrid approach was applied in order to increase the classification performance in the identification of lung sounds. Lung sound classes consist of normal, wheezes, crackles, and crackles plus wheezes. A pretrained CNN model, which utilizes spectrogram images as input, was used for deep-feature extraction. In the classification stage, Linear Discriminant Analysis (LDA) classifier was employed together with the Random Subspace Ensembles (RSE) method.

The remaining sections of this paper are organized as follows: with Methodology in Section 2, Experimental Works in Section 3, and Conclusions in Section 4. The Methodology section provides information about both the framework of the proposed method and the spectrogram image, CNNs, LDA, and RSE methods. In the Experimental Works section, the proposed method is evaluated using a robust dataset, with experimental results shown in tabular format. In the Conclusions section, the experimental results are interpreted according to performance criteria and other methods that have used the same dataset.

II. THE METHODOLOGY

The framework of the proposed method is illustrated in Fig. 1. In the preprocessing stage, spectrogram images are constituted using the lung sounds. Because of the varying frequencies of the sample lung sounds, the window size and overlap of the spectrograms is selected according to sampling frequency values. The spectrogram data is then transformed to image format using Viridis Color Map. In this way, spectrogram images are created and saved in the folders. The pretrained CNN model, which is illustrated in Fig. 2, is constituted by training with a particular part of the spectrogram images. The deep features are extracted from the fully-connected layer of the pre-trained CNN model. To increase classification performance, while doing the down-sampling process by the pooling layers, it is planned that the average pooling process of the characteristic information, which is eliminated by the max-pooling process, would be captured. As a reason, the pooling structure used in the pre-trained CNN model is parallel-connected the max-pooling layer to the average pooling layer. In order to test the classification performance of the proposed method, the deep features are conveyed to the LDA classifier. To further increase the classification performance, the RSE method is applied to the LDA classifier.

A. SPECTROGRAM IMAGES

A spectrogram is a visual process that illustrates the power, or loudness, of a signal over time at different frequencies within a certain waveform. The spectrogram also shows how energy levels vary over time. The Short-Time Fourier Transform (STFT) formulation is as shown in Equation 1:

$$F(n, \omega) = \sum \infty \; x(i)\omega(n - i)e{-}jwn \qquad (1)$$

where x(i) is input, and ω(i) is a window function (e.g., hamming window and rectangular window) that is generally centered at time n. A spectrogram can be expressed as the squared magnitude of the STFT. The spectrogram images are constituted with Viridis Color Map, which is a homogeneous mapping that utilizes colors changing from blue to green to yellow [16], [17].

B. CNNs

The main CNN processes consists of the forward- propagation and back-propagation. The learning parameters of the forward-propagation are optimized by way of the back-propagation techniques [18]. The forward-propagation consists of numerous convolutional layers, plus one or more fully-connected layers. The aim of the convolutional layers is to extract apparent attributes from the input signal as it is conveyed through the layers.

In general, the low-level features are extracted by the first convolutional layer, whilst the more complicated features are extracted by the subsequent convolutional layers. The training process of the CNNs is as follows.
The convolutional, batch normalization, activation, pooling and softmax layers form the forward-propagation stage.
The basic aim in the convolution stage is to assign local connections of features from the prior layers and map their data to the feature maps. The convolution operation of the 2D data is shown in Equation 2:

$$yn = \sum i\ yn{-}1 * \omega n + bn \qquad (2)$$
$$i \qquad i \qquad ij \qquad i$$

where yn–1 is the input data or the previous convolutional output, ωn is n-th weight matrix, and bn is n-th bias vector.

Before conveying the data in the convolutional layer to a nonlinear function, the data is prepared not to show an abnormal distribution with Batch Normalization (BN) layer, which prevents gradient vanishing during the training stage.
Thus, the learning parameters are optimized in order to speed up convergence by protecting a state of greater gradient at all times. Besides, the BN layer is employed to decrease the level of noise [19]. The operation of the BN layer is expressed as shown in Equations 3-6:

$$m_b = \frac{1}{k}\sum_{i=1}^{k} x_i \qquad (3)$$

$$v_b = \frac{1}{k}\sum_{i=1}^{k} (x_i - m_b)^2 \qquad (4)$$

$$\hat{x}_i = \frac{x_i - m_b}{\sqrt{v_b^2 + \epsilon}} \qquad (5)$$

$$y_i^n = c\hat{x}_i + d \qquad (6)$$

where x_i is the input, m_b is the mini-batch mean, v_b is the mini-batch variance, k is the input size, and ϵ is the small constant. Scale and shift factors are represented as c and d, respectively. These factors are learnable parameters adjusted to the most convenient values during the training process. The value y_i^n is the i-th output of n-th BN layer.
The Rectified Linear Unit (ReLU), which is the most used activation in the CNNs, prevents gradient explosion and gradient disappearance problems within the sigmoid activation function. The ReLU activation function is as shown in Equation 7:

$$rn = \max(0, yn) \qquad (7)$$
$$i \qquad i$$

where rn is the i-th output of n-th ReLU layer. The down-sampling operation, which decreases the matrix size, is applied by the pooling layer, which in turn reduces the computation cost and prevents over-fitting [20]. The most common pooling layers are maximum and average pooling layers. The calculation of the pooling operation is expressed as shown in Equation 8:

$$pn = average\ or\ maximum\ operation\{rn\} \quad (8)$$

In the first fully-connected layer, the matrices conveyed scores. The softmax is the only operation that transforms the output of the neural network via probability distribution, giving the possibility of different classes. The softmax operation is as shown in Equation 9:

$$y_i^n = \sigma(x_i) = \frac{e^{fi}}{\sum_{j=1}^{K} e^{xj}} \quad (9)$$

where fi is the i-th output of the previous fully-connected layer, ynsymbolizes the i-th predicted score of the CNNs, and K is the number of predicted scores of the CNNs.

In the back-propagation stage, the cross-entropy function is used for the CNNs. This provides information about the distance between the predicted values dispersion and the ground truth dispersion. Calculation of the cross-entropy function can be expressed as Equation 10:

$$H(y_i^t, y_i^n) = -\sum_{j=1}^{K} y_i^t log y_i^n \quad (10)$$

where yt symbolizes the ground truth values, and $yn\ i$ i symbolizes the predicted values. With the L2 regularization factor, the cross entropy is rearranged as in Equation 11:

$$L(m, b) = H(yt, yn) + a \sum m\ m2 \quad (11)$$

i i

where α is the L2 regularization coefficient, b is the bias values and $L(\omega, b)$ is the loss function. All learning parameters of the CNNs are updated through the optimizationmethods, such as the Stochastic Gradient Descent Momentum(SGDM) and the Adam. Optimization of the learning parameters can be expressed as shown in Equation 12:

$$m_{ij}^{n} = m_{ij}^{n-1} + \delta \frac{\partial}{\partial m_{ij}} L(m, b) \quad (12)$$

where mn is the updated learning parameter, $mn–1$ is the ij ij from the previous layers are flattened and connected to other fully-connected layers. The CNNs structure from the fully- connected layer to the classification layer is the same as for multilayer perceptron (MLP).

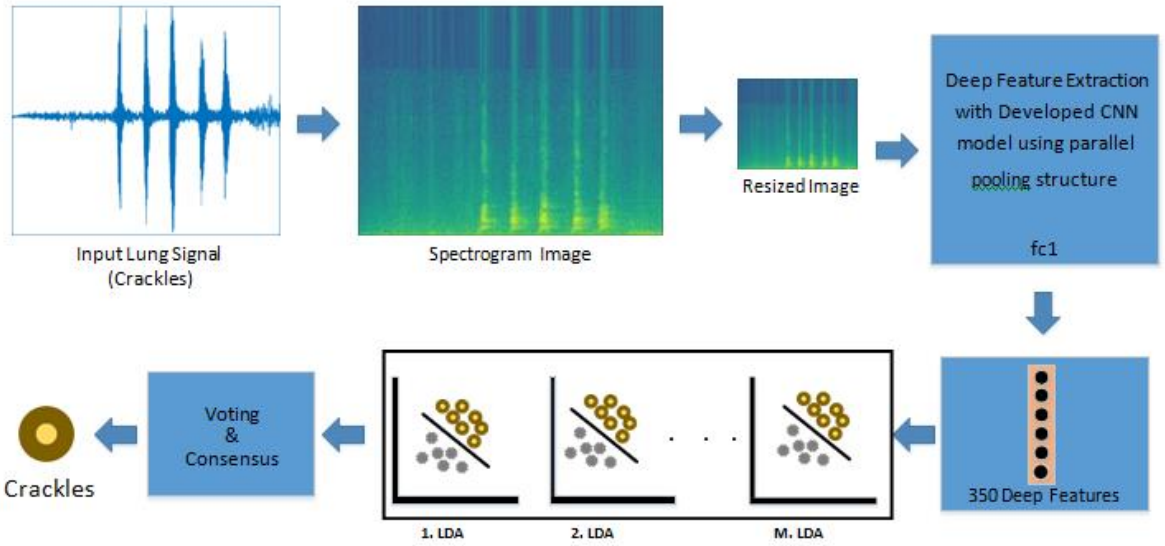
The values from the previous fully-connected layer are used as the input to the softmax layer. The softmax operation can be defined as a learning method used to adjust classification previous learning parameter, and δ is the learning rate.

C. LINEAR DISCRİMİNANT ANALYSİS

Given an X matrix (s×t), which is processed as c column vectors x1, x2,…, xs (x ϵ ℜt), each column represents a data point and each row represents a certain feature [21]. With the linear transformation matrix B ϵ ℜs×u, the feature set (y ϵ ℜl) can be calculated as shown in Equation 13:

$$\boldsymbol{B}: \boldsymbol{x} \in \mathfrak{R}\boldsymbol{s} \rightarrow \boldsymbol{y} = \boldsymbol{BTx} \in \mathfrak{R}\boldsymbol{u} \quad (13)$$

The outcome matrix (Y ϵ ℜu×n) includes u rows which leads to the u-dimensional decreased space, and each data point consist of u features.



M Random Subspaces LDA Learners

FIGURE 1. Framework of the proposed method.

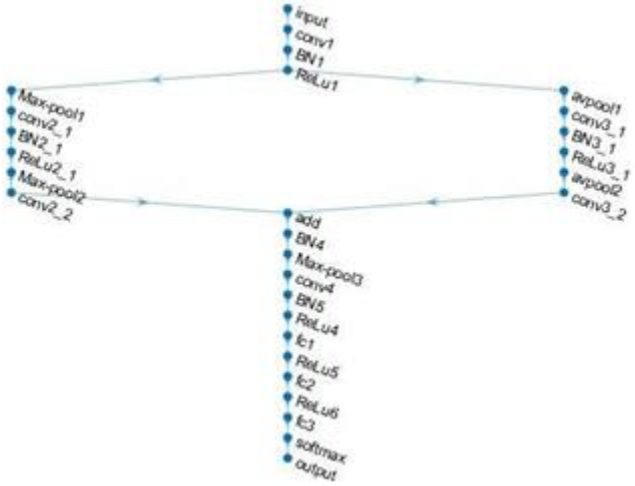


FIGURE 2. Illustration of the proposed CNN using parallel pooling structure

Given the within-classes scatter matrix Sm, the between- classes scatter matrix Sn, and the scatter matrices S, $\boldsymbol{SL}$ and $\boldsymbol{L}$ correspond to the between-classes scatter matrix and within-classes scatter matrix in the lower-dimensional space.

$\boldsymbol{SL}$, $\boldsymbol{SL}$ and the scattering matrix criterion ($\boldsymbol{J}(\boldsymbol{B})$) is

$\boldsymbol{m}$ $\boldsymbol{n}$

calculated by the linear transformation B using Equations 14-16:

$$\mathbf{B} = \arg\max_{\boldsymbol{B}}[(\boldsymbol{B}^T\boldsymbol{S}_m\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{S}_n\boldsymbol{B}] \quad (17)$$

For each Gaussian class using the same covariance matrix, the LDA is similar to the Bayesian classifier, with only a variation in its threshold value.

D. RANDOM SUBSPACE ENSEMBLES

The RSE is a method used to boost the performance of classifiers. The method is grounded on a stochastic operation that randomly chooses a number of components of the learning model in creating each classifier [22]. In the RSE method, the training data is split into random subspaces. The most convenient subspace class membership is assigned by the learner algorithm of the classifiers. Then, class memberships conveyed from each subspace learner is assembled in a class vector. The predicted scores are stated by the highest average score of the class vector. For the LDA classifier, the RSE method is applied as follows:

- Item 1: Choose without changing random data of the K-size from training data (K < N).
- Item 2: Train an LDA learner using only the predictors.
- Item 3: Apply Item 1 and Item 2 until there are M LDA learners.
- Item 4: Assemble prediction values of the LDAlearners.
- Item 5: Classify the test dataset with the highest average value.

$\boldsymbol{SL}$ $\boldsymbol{BTS}$ $\boldsymbol{B}$

To optimize B in the scattering matrix criterion, $\boldsymbol{SL}$ should be maximized while $\boldsymbol{SL}$ should be minimized. The optimal B in the LDA classifier is computed as shown in Equation 17:

The representation of the RSE, which employs the LDA classifier, is shown in Fig. 3, where K is the dimension of subspaces, d is the training samples selected as random, and M is the number of LDA learners.

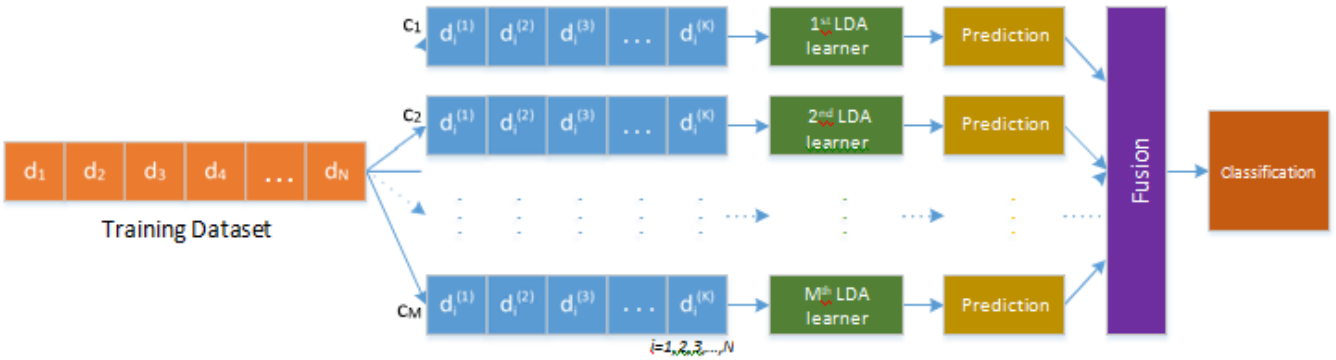


FIGURE 3. Representation of random space ensemble with LDA.

III. EXPERIMENTAL WORKS

A. DATASETS

The proposed method was evaluated against the ICBHI 2017 Challenge dataset, which consists of 920 lung sounds. The lung sounds were recorded using three different digital stethoscopes, with sampling frequencies of 4 KHz, 10 KHz, and 44.1 KHz, respectively. The classes of the ICBHI 2017 Challenge dataset consist of crackles, wheezes, normal, and wheezes plus crackles. A 20-second sound file may include one or more class tag since the sound files are separated into cycles. The cycle breakdown for a sound file is presented in Table 1, with columns shown as cycle index, start time, end time, and values for both crackles and wheezes. The wheeze and crackle values for crackles, wheezes, normal, and wheezes plus crackles tags are 1-0, 0-1, 0-0, and 1-1, respectively. According to the class tags, the total number of the cycles is given in Table 2.

TABLE I CYCLE INFO FOR A SOUND FILE

Cycle	Start time	End time	Crackle value	Wheeze value	I
1	0.804	3.256	0	0	e p
2	3.256	5.566	0	0	of
3	5.566	7.851	0	1	c e
4	7.851	10.054	0	1	c
5	10.054	12.066	1	0	pr
6	12.066	14.47	1	0	c
7	14.47	16.696	1	1	
8	16.696	18.887	1	1	
9	18.887	19.792	1	1	

TABLE II CYCLE BREAKDOWN OF ICBHI 2017 CHALLENGE DATASET

Number of cycles	Total
With crackles	1,864
With wheezes	886
With crackles + wheezes	506
Normal cycles	3,642
Total number of cycles	6,898

B. EVALUATION METHOD AND CRITERIA

In the ICBHI 2017 Challenge dataset, 90% of the data were parated for the purposes of training and validating the oposed CNN. The remaining 10% of the dataset was used r the LDA classifier combined with the RSE method in the lassification stage. Classification performance was valuated with 10-fold cross-validation. The performance riteria were selected as accuracy, specificity, sensitivity, ecision, and F-score. Each of these criteria, along with the onfusion matrix, are shown in Equations 18-22:

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (18)$$

$$Specificity = \frac{TN}{TN+FP} \quad (19)$$

$$Sensitivitiy = \frac{TP}{TP+FN} \quad (20)$$

$$Precision = \frac{TP}{TP+FP} \quad (21)$$

$$F-score = 2x\frac{PrecisionxSensitivity}{Precision+Sensitivity} \quad (22)$$

C. EXPERİMENTAL SETUP AND RESULTS

The experimental works were applied using MATLAB (R2019a) software installed on a computer with an Intel(R) Core(TM) i7-5500 CPU @2.4 GHz with 4 core, NVİDİA GeForce 840M GPU, and 8 GB RAM.

In the proposed method, the spectrogram operation was applied to the lung sound data. Window size and overlap, which are spectrogram parameters, were adjusted to the sampling frequencies. According to the 4 KHz, 10KHz, and 44.1 KHz sample frequencies, window size and overlap were selected as 64-8, 128-16, and 524-64, respectively. In these selections, the resolution of the spectrogram images is the most significant factor, as the spectrogram images are used for the input for training the proposed CNN. The layer parameters of the proposed CNN shown in Fig. 2 are presented in Table 3.

TABLE III LAYER PARAMETERS OF PROPOSED CNN

Layers	Filter numbers	Filter size	Stride
conv1	64	5×5	1
conv2_1	32	3×3	1
conv2_2	32	3×3	1
conv3_1	32	3×3	1
conv3_2	32	3×3	1
conv4	32	3×3	1
all pooling layer	-	2×2	2

The initial learning rate, max-epochs and validation frequency, which are the training option parameters, were selected as 0.005, 12, and 30, respectively. The Adam optimizer is employed for the training process. According to the iterations, the training and validation accuracy and the loss deviation of the proposed CNN is illustrated in Fig. 3. At the end of 519 iterations, while the training accuracy was around 60%, the validation accuracy was found to be 49.78%.

The first fully-connected layer (fc1) of the proposed CNN was used to extract 350 deep features, which were then evaluated using the LDA classifier and the RSE method with the subspace dimension adjusted to 64.

The classification results with 10-fold cross-validation are presented in the confusion matrix shown as Fig. 5. The best accuracy score achieved was 83.2% with the normal class, whilst the worst accuracy score achieved was 40.4% with the wheezes class. The average accuracy score was 71.15%.

The results of the other performance criteria, specificity, sensitivity, precision, and F-score, are presented in Table 4, with the best score for each class shown as bold typeface.

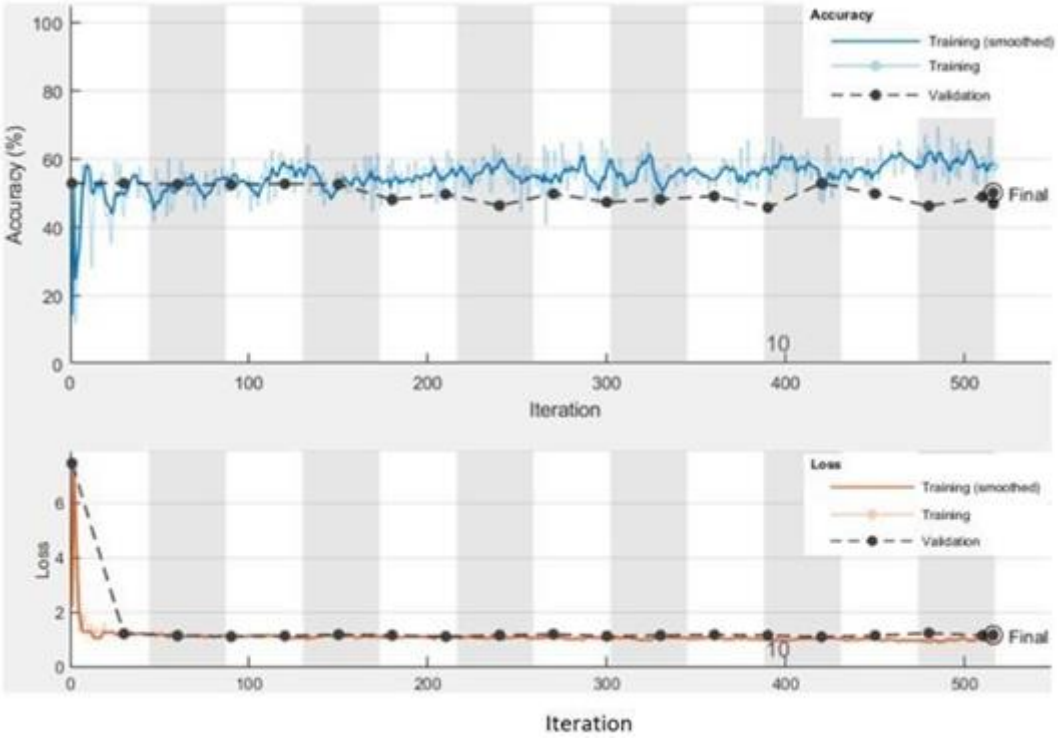


FIGURE 4. Training, validation and loss deviation against iterations.

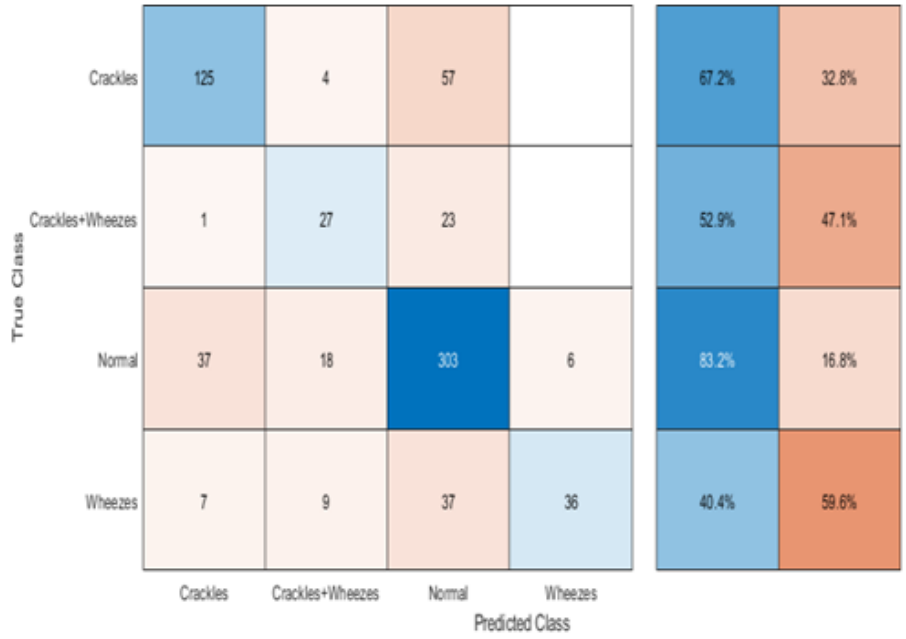


FIGURE 5. Confusion matrix for the proposed method.

The best specificity and precision values were achieved with the wheezes class, and the best sensitivity and F-score values were achieved with the normal class. For the proposed method, the classifier was selected by using the classification accuracy criterion. In Table 5, the classification results are shown for different classifiers, with the best classification accuracy achieved with the LDA classifier using the RSE method. In Table 6, the classification performance of the proposed CNN is compared with the other popular pretrained CNN models, and the CNN structures obtained when there is only one parallel connection branch, as shown in Fig. 2.

For deep features extracted from CNN models, classifiers that give the best performance are used, and among the popular pretrained CNN models, the VGG16 model yields the best level of accuracy at 65.4%. While the proposed CNN model only has max-pooling and average-pooling branches, the accuracy results of the proposed method were shown to be 67.2% and 68.6%, respectively. With the parallel-pooling structure in the proposed CNN, the best accuracy level achieved was 71.15%.

In Table 7, the proposed method is compared with other methods using the same dataset based on their accuracy s

TABLE IV RESULTS OF OTHER PERFORMANCE CRITERIA

Class / **Performance Criteria**	Specificity	Sensitivity	Precision	*F*-score
Crackles	0.89	0.67	0.74	0.70
Wheezes	**0.99**	0.40	**0.86**	0.55
Crackles+Wheezes	0.94	0.53	0.47	0.50
Normal	0.62	**0.83**	0.72	**0.77**
Average Score	0.86	0.61	0.69	0.65

TABLE V ACCURACY SCORES FOR OTHER CLASSIFIERS

Classifiers	**(Acc%)**
LDA	57.0
Decision Tree	65.4
SVM	68.8
KNN	68.0
KNN-RSE	66.5
LDA-RSE	**71.15**

TABLE VI ACCURACY RESULTS FOR CNN MODELS

CNN model	**Classifier**	**% Acc**
GoogleNet	SVM	52.3

Squeezenet	KNN	55.8
Inceptionresnetv2	KNN-RSE	57.6
ResNet101	LDA-RSE	58.4
ResNet-50	SVM	59.1
AlexNet	SVM	60.7
VGG19	SVM	62.3
VGG16	SVM	65.4
Only average-pooling	LDA-RSE	67.2
Only max-pooling	LDA-RSE	68.6
Proposed CNN (parallel-pooling)	LDA-RSE	**71.15**

In [23], features were extracted using MFCC, and evaluated with the Hidden Markov Model classifier, and achieved a best accuracy level of 39.56%. In [24], low-level features were used for the feature extraction, and the features then conveyed to the Decision Tree classifier, and lung sounds were classified to an accuracy of 49.62%. In [25], the wavelet decomposition and STFT were combined as a feature set, producing a best accuracy level of 57.88% using the SVM classifier. In [26], two methods were proposed for lung sound classification. First, lung sounds were classified with the transfer learning technique, trained by applying fine-tuning to the pretrained VGG16 model, and achieved a best accuracy level of 63.09%. In the second method, deep features were extracted from the fully-connected layers of the pretrained VGG16 model, realizing a best accuracy level of 65.50% with the SVM classifier.

IV. CONCLUSION

In this paper, lung sounds are used to classify pulmonary disorders. In the literature, traditional machine learning techniques are generally used for lung sound classification, although more recently, techniques based on deep learning have started to be used for classification performance. Popular pretrained CNN models such as VGG16 and AlexNet, have mostly given good results for image recognition and also for some sound classification applications; however, sound characteristics are not fully represented since these CNN models have not been trained on sound datasets. Hence, the proposed CNN model was trained with spectrogram images based on lung sounds. In addition, the parallel-pooling structure was employed in order to boost classification performance in the proposed CNN architecture. Then, deep features were extracted from the first fully-connected layer of the proposed CNN. The deep features are employed by using different classification algorithms, with the best obtained result being 71.15% with the LDA-RSE classifier. Performance of the proposed CNN model was compared with the other popular pretrained CNN models, and the best classification accuracy was achieved by using the proposed CNN model. In addition, the accuracy score of the proposed method resulted in an improvement increase of 5.75%, when compared to the other best result methods using the same dataset.

TABLE VII ACCURACY RESULTS FOR OTHER METHODS USING SAME DATASET

Authors	Methodology	% Acc
Jakovljević et al. [23]	MFCC, Hidden Markov Model	39.56
Chambres et al. [24]	Low Level Feature, Decision Tree	49.62
Serbes et al. [25]	STFT+Wavelet, SVM classifier	57.88
Demir et al. [26]	Transfer learning with CNN Model and softmax classifier	63.09
Demir et al. [26]	Deep Feature with VGG-16 CNN model and SVM classifier	65.50
Proposed method	Deep feature with CNN & LDA classifier with RSE method	**71.15**

V. REFERENCES

[1]. WHO, WHO | World Health Statistics 2015. World Health Organization, 2016.

[2]. G. C. Chang and Y. P. Cheng, "Investigation of noise effect on lung sound recognition," Proceedings of the 7th International Conference on Machine Learning and Cybernetics, ICMLC, 2008, vol. 3, pp. 1298–1301, doi: 10.1109/ICMLC.2008.4620605.

[3]. G. C. Chang and Y. F. Lai, "Performance evaluation and enhancement of lung sound recognition system in two real noisy environments," Comput. Methods Programs Biomed., vol. 97, no. 2, pp. 141–150, 2010, doi: 10.1016/j.cmpb.2009.06.002.

[4]. S. Reichert, R. Gass, C. Brandt, and E. Andrès, "Analysis of Respiratory Sounds: State of the Art," Clin. Med. Circ. Respirat. Pulm. Med., vol. 2, 2008, doi: 10.4137/ccrpm.s530.

[5]. N. Sengupta, M. Sahidullah, and G. Saha, "Lung sound classification using cepstral-based statistical features," Comput. Biol. Med., vol. 75, pp. 118–129, 2016, doi: 10.1016/j.compbiomed.2016.05.013.

[6]. S. Içer and Ş. Gengeç, "Classification and analysis of non-stationary characteristics of crackle and rhonchus lung adventitious sounds," Digit. Signal Process. A Rev. J., vol. 28, no. 1, pp. 18–27, 2014, doi: 10.1016/j.dsp.2014.02.001.

[7]. R. Naves, B. H. G. Barbosa, and D. D. Ferreira, "Classification of lung sounds using higher-order statistics: A divide-and-conquer approach," Comput. Methods Programs Biomed., vol. 129, pp. 12– 20, 2016, doi: 10.1016/j.cmpb.2016.02.013.

[8]. G. Serbes, C. O. Sakar, Y. P. Kahya, and N. Aydin, "Pulmonary crackle detection using time-frequency and time-scale analysis," Digit. Signal Process. A Rev. J., vol. 23, no. 3, pp. 1012–1021, 2013, doi: 10.1016/j.dsp.2012.12.009.

[9]. F. Jin, F. Sattar, and D. Y. T. Goh, "New approaches for spectro- temporal feature extraction with applications to respiratory sound classification", Neurocomputing, vol. 123, pp. 362–371, 2014, doi: 10.1016/j.neucom.2013.07.033.

[10]. M. Bahoura, "Pattern recognition methods applied to respiratory sounds classification into normal and wheeze classes," Comput. Biol. Med., vol. 39, no. 9, pp. 824–843, 2009, doi: 10.1016/j.compbiomed.2009.06.011.

[11]. Y. P. Kahya, M. Yeginer, and B. Bilgic, "Classifying respiratory sounds with different feature sets," Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings, 2006, pp. 2856–2859, doi: 10.1109/IEMBS.2006.259946.

[12]. S. Jayalakshmy and G. F. Sudha, "Scalogram based prediction model for respiratory disorders using optimized convolutional neural networks," Artif. Intell. Med., vol. 103, s. 101809, 2020, doi: 10.1016/j.artmed.2020.101809.

[13]. M. Valenti, A. Diment, G. Parascandolo, S. Squartini, and T. Virtanen, "{DCASE} 2016 Acoustic Scene Classification Using Convolutional Neural Networks," Proceedings of the Detection and Classification of Acoustic Scenes and Events 2016 Workshop (DCASE2016), 2016, pp. 95–99, doi: 10.1111/j.1558-5646.2010.01180.x.

[14]. M. Aykanat, Ö. Kilic, B. Kurt, and S. Saryal, "Classification of lung sounds using convolutional neural networks," EURASIP J. Image Video Process., vol. 2017, no. 65, pp. 1-9, 2017.

[15]. D. Bardou, K. Zhang, and S. M. Ahmad, "Lung sounds classification using convolutional neural networks," Artif. Intell. Med., vol. 88, no. 1, pp. 58–69, 2018, doi: 10.1016/j.artmed.2018.04.008.

[16]. K. Yanai and Y. Kawano, "Food image recognition using deep convolutional network with pre-training and fine-tuning", 2015 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2015, 2015, pp. 1–6, doi: 10.1109/ICMEW.2015.7169816.

[17]. Y. Zhou, H. Nejati, T. T. Do, N. M. Cheung, and L. Cheah, "Image- based vehicle analysis using deep neural network: A systematic study," International Conference on Digital Signal Processing, DSP, 2016, vol. 0, pp. 276–280, doi: 10.1109/ICDSP.2016.7868561.

[18]. S. Dong, W. Wu, K. He, and X. Mou, "Rolling bearing performance degradation assessment based on improved convolutional neural network with anti-interference," Meas. J. Int. Meas. Confed., vol. 151, p. 107219, 2020, doi: 10.1016/j.measurement.2019.107219.

[19]. S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 32nd Int. Conf. Mach. Learn. ICML 2015, vol. 1, pp. 448–456, 2015.