applied sciences

MDPI

Article

Respiratory Sound Classification by Applying Deep Neural Network with a Blocking Variable

Runze Yang [1], Kexin Lv [1], Yizhang Huang [1], Mingxia Sun [2], Jianxun Li [1] and Jie Yang [1,*]

[1] Institute of Image Processing and Pattern Recognition, Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China; runze.y@sjtu.edu.cn (R.Y.); kelen_lv@sjtu.edu.cn (K.L.); hyizhang@alumni.sjtu.edu.cn (Y.H.); lijx@sjtu.edu.cn (J.L.)
[2] Medtronic Technology Center, Shanghai 201114, China; michelle.sun@medtronic.com
* Correspondence: jieyang@sjtu.edu.cn

Abstract: Respiratory diseases are leading causes of death worldwide, and failure to detect diseases at an early stage can threaten people's lives. Previous research has pointed out that deep learning and machine learning are valid alternative strategies to detect respiratory diseases without the presence of a doctor. Thus, it is worthwhile to develop an automatic respiratory disease detection system. This paper proposes a deep neural network with a blocking variable, namely Blnet, to classify respiratory sound, which integrates the strength of the ResNet, GoogleNet, and the self-attention mechanism. To solve the non-IID data problem, a two-stage training process with the blocking variable was developed. In addition, the mix-up data augmentation within the clusters was used to address the imbalanced data problem. The performance of the Blnet was tested on the ICBHI 2017 data, and the model achieved 79.13% specificity and 66.31% sensitivity, with an average score of 72.72%, which is a 4.22% improvement in the average score and a 12.61% improvement in sensitivity over the state-of-the-art results.

Keywords: deep neural network; non-IID problem; respiratory sound classification; signal processing

Citation: Yang, R.; Lv, K.; Huang, Y.; Sun, M.; Li, J.; Yang, J. Respiratory Sound Classification by Applying Deep Neural Network with a Blocking Variable. *Appl. Sci.* **2023**, *13*, 6956. https://doi.org/10.3390/app13126956

Academic Editors: Qingyu Zhao, Daniel Moyer and Magdalini Paschali

Received: 16 April 2023
Revised: 5 June 2023
Accepted: 6 June 2023
Published: 8 June 2023

1. Introduction

Respiratory diseases ranked the fourth leading cause of death worldwide by the world health organization (https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death (accessed on 9 December 2020)). Typical respiratory diseases include asthma, chronic obstructive pulmonary disease (COPD), and lower respiratory tract infection. These diseases can threaten people's lives if treatment is not provided on time. Thus, the early detection of the diseases is crucial. However, the resources of experienced doctors are limited. Through Salvatore's research [1], he discovered hospital trainees and medical students made the wrong diagnosis on almost half of the pulmonary sounds. With the outbreak of COVID-19, the lack of medical resources is even more acute [2]. The prevalence of respiratory diseases has increased due to the long-term sequelae resulting from COVID-19 [3]. Thus, it is meaningful to develop an automatic respiratory disease detection system. The system can help improve the correct diagnosis of respiratory diseases for hospital trainees and enable the patients to have a quick diagnosis anywhere, even at home. In general, a respiratory sound could be classified as a normal or adventitious sound. Crackles and wheezes are two of the most common adventitious sounds. A wheeze is a continuous adventitious sound, which is caused by the limitation of the airflow due to the tightening of the airway [4]. Wheezing may occur at either the inspiration stage or the expiration stage, but mostly at the expiration stage. A wheeze is a high-pitch sound, and its frequency range is generally between 100 and1000 Hz, mostly larger than 400 Hz, and its minimum duration is between 80 ms and 100 ms [5]. The most common diseases associated with wheezing are asthma and COPD. A crackle is a discontinuous sound, which is caused by air bubbles in the bronchi, and it is a short, explosive sound [6]. The sounds can be

found mostly at the early stages of inspiration but are also audible at the expiatory stage. A crackle is a low-pitch sound, and its frequency is between 100 and 500 Hz, mostly at 350 Hz, with a duration usually around 15 ms [7]. The most common diseases associated with a crackle sound is chronic bronchitis, bronchiectasis, and COPD. In addition, the lungs are much larger than the heart, so the respiratory sounds should be recorded at multiple sites of both lungs for an accurate diagnosis. Those positions are the trachea, anterior, posterior and lateral.

The ICBHI 2017 database [8] is the largest open respiratory sound database published by the International Conference on Biomedical Health Informatics (ICBHI), which has attracted numerous teams to develop their own automatic respiratory sound classificaition system. In the previous research, both traditional machine learning and deep learning fields have published a large number of papers to tackle the classification task of the ICBHI 2017 respiratory sound database. These authors proposed three different traditional machine learning methods, including the Hidden Markov Model with the Gaussian Mixture model [9], Boosting Decision Tree [10], and Support Vector Machine [11], to conduct the classification. With the rise of computation power, deep learning became popular and achieves a better performance. The authors of [12] used the deep recurrent neural network with a noise masking model, Ref. [13] introduced a hybrid CNN-RNN and patient-specific model, and Ma et al. [14] proposed a Bi-resnet to implement the classification. Ref. [15] developed a lung network with a non-local layer, and proposed a special mix-up data augmentation to increase the data size. Ref. [16] introduced device-specific fine-tuning and also mix-up data augmentation to improve the performance of their Respire network. In all these deep learning models, the popular input features of the deep neural network include the short-term Fourier transform [12,13], mel-spectrogram [16], and wavelet transform [14].

Based on our research, the short Fourier transform and wavelet transform achieve the best performance as the input of the deep neural network. Thus, we used both of them to pre-process the data, and proposed the Blnet to conduct the classification, in which we developed several techniques to solve the limited, imbalanced, and non-IID data problem to improve the performance of our model. The main contributions of this paper can be summarized in the following four points:

- A small deep neural network with a diverse structure called Blnet is proposed by integrating the strength of ResNet [17], GoogleNet [18], and the self-attention mechanism [19].
- A simplified loss function that enables the network to handle a four-class classification with only two outputs.
- Mix-up data augmentation within the clusters is suggested to address the imbalanced data problem.
- A two-stage training process with the blocking variable is developed to address the not-independently and identically distributed (non-IID) data problem.

2. Pre-Processing the Sound

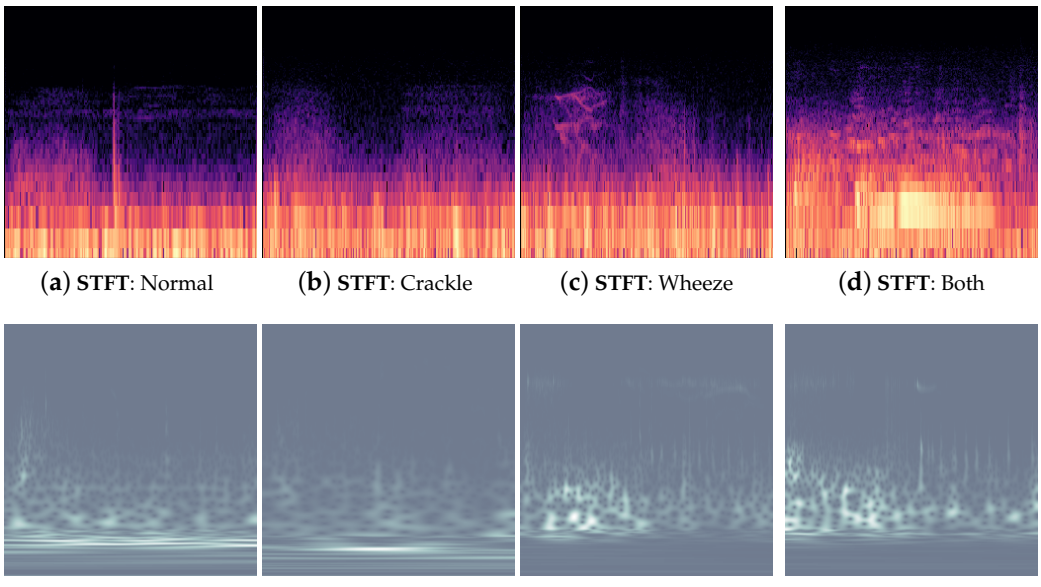
2.1. ICBHI 2017 Data

The ICBHI 2017 [8] data set is used in this paper to develop our automatic respiratory sound classification system. The data set was collected from 126 patients and includes 920 audio samples. The summation of the 920 audio samples is 5.5 h long. They were clipped into 6898 respiratory cycles, which include 3643 normal cycles, 1864 cycles with crackles, 886 cycles with wheezes, and 506 cycles with both crackles and wheezes. Those audio samples were collected in seven different chest locations, and they were the trachea (Tc), anterior left (Al), anterior right (Ar), posterior left (Pl), posterior right (Pr), lateral left (Ll), and lateral right (Lr). Four types of equipment were used to collect the data, and they were a AKG C417L Microphone (AKGC417L), a 3M Littmann Classic II SE Stethoscope (LittC2SE), a 3M Litmmann 3200 Electronic Stethoscope (Litt3200), and a WelchAllyn Meditron Master Elite Electronic Stethoscope (Meditron). The data were also collected under two types of acquisition modes, which were single-channel (sc) and multi-channel (mc). In previous research, there were two popular ways to split the ICBHI 2017 respiratory

data into the train and test set. The first one is the official train–test set split, and the second way is randomly separating the train set and test set according to the ratio of 8:2. The proportion of train set and test set in the official train–test split is about 6:4. We have carried out research on both splits to test our model's performance.

2.2. Short Fourier Transform

Firstly, 920 examples of annotated respiratory sound data were clipped into 6898 respiratory cycles. Since the ICBHI 2017 data were collected in different hospitals with different environment noises, including the human voice and heartbeat, we used a third-order Butterworth band-pass filter to remove the environment noise sound below 10 Hz or larger than 4000 Hz [20], and normalized the signal to 0 and 1. The short-time Fourier transform (STFT) method [21] was used to extract the features of the respiratory sounds in the time and frequency domain. However, the data were collected in different hospitals with different types of equipment. As a result, the ICBHI 2017 sound data have two sampling frequencies, 4000 Hz and 44,100 Hz. The length of a respiratory cycle may vary between 0.2 s and 16.2 s for ICBHI 2017 data, with a mean equal to 2.7 s. Thus, if we let both the hop length and the window length of the STFT filter be proportional to the sampling frequency, the output spectrogram would have a distinct shape for sound data with the sampling frequency equal to 4000 Hz and 44,100 Hz. Instead, we decided to let the hop length of the filter be equal to 0.01 times the sampling frequency and the filter's window length equal to 1024. This ensures that the shapes of the spectrograms are similar before they are stretched and compressed into the same shape. The shape of the input image of the neural network is 128 pixels times 128 pixels. Examples of the input images are shown at the top of Figure 1.



(**a**) **STFT**: Normal (**b**) **STFT**: Crackle (**c**) **STFT**: Wheeze (**d**) **STFT**: Both

(**e**) **WT**: Normal (**f**) **WT**: Crackle (**g**) **WT**: Wheeze (**h**) **WT**: Both

Figure 1. Two types of input images of the deep neural network.

2.3. Wavelet Transform

The ICBHI 2017 lung sound data are fluctuating and noisy; therefore it is hard to capture the global and local features of the time and frequency domain at the same time. There is a trade-off between the time and frequency domain in the STFT. Thus, we also adopted the wavelet transform (WT) [22] to improve the diversity of the input. We adopted the same clipping and filtering process as we did for the STFT. Then the continuous wavelet transform was applied to the signals and we transferred the signals to the scalograms. The scalograms could identify the features under different frequency domains and are less sensitive to noise. The Morlet wavelet kernel was selected after the experiments. Finally,

the scalograms were stretched and compressed into the same shape. The resolutions of the scalogram and the spectrogram were the same. Examples of the input images are shown at the bottom of Figure 1.

2.4. Data Augmentation within the Clusters

The ICBHI 2017 data set, similar to many other medical data sets, is also characterized by an imbalanced data problem. The proportions of Normal, Crackle, Wheeze, and Both data are 53%, 27%, 13%, 7%, respectively. To address this problem, we decided to use mix-up data augmentation [23] to increase the size of the adventitious sounds. We let two randomly selected Wheeze data sets mix together to generate a new Wheeze data set, and two randomly chosen Crackle data sets mix together to generate a new Crackle data set. The data augmentation for Wheeze data was carried out twice, and no data argumentation was carried out for Both data, as this would likely lead to overfitting of the model by our experiment. Furthermore, the data augmentation was carried out within seven clusters, which represent the seven chest locations where the data were collected from. This was because the respiratory sounds did not follow the same distribution between different clusters. We show the final training set after the data augmentation for the 8:2 train–test split as an example in Table 1. The same technique was also applied to the ICBHI official train–set split. The mathematical details are shown in the following.

$$\begin{aligned} \hat{X_k} &= \lambda X_{ki} + (1-\lambda) X_{kj}, \\ \hat{Y_k} &= \lambda Y_{ki} + (1-\lambda) Y_{kj}, \end{aligned} \tag{1}$$

where the integer $k \in [1, 7]$ is the index of the cluster, representing the 'Tc', 'Al', 'Ar', 'Pl', 'Pr', 'Ll', and 'Lr', respectively; X_{ki} and X_{kj} represent two randomly selected input feature from the cluster k; and $\hat{X_k}$ represents the new augmented data within the cluster k. $\hat{Y_k}, Y_{ki}, Y_{kj}$ are the labels of new augmented data and the selected feature data, which will be identical, and λ is generated from a Beta distribution with $\alpha = \beta = 1$.

Table 1. The data augmentation is carried out within the clusters. The cluster1, cluster2, cluster3, cluster4, cluster5, cluster6, cluster7 represent the 'Tc', 'Al', 'Ar', 'Pl', 'Pr', 'Ll', 'Lr', respectively.

	Cluster1		**Cluster2**		**Cluster3**		**Cluster4**		**Cluster5**		**Cluster6**		**Cluster7**		**Total**	
Before/After	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**
Normal	530	530	580	580	566	566	333	333	427	427	202	202	271	271	2910	2910
Wheeze	89	178	225	450	222	444	308	616	202	404	188	376	264	528	1498	2996
Crackle	99	297	114	342	150	450	106	318	100	300	51	153	75	225	695	2085
Both	20	20	87	87	65	65	69	69	80	80	38	38	56	56	415	415

3. The Architecture of the Network

3.1. The Block-I, Block-II, and the Self-Attention Block

We proposed a network called the Blnet, which consists of three important blocks, the Block-I, Block-II, and the Self-attention block. The architecture of the Blnet without the blocking variable is shown in Figure 2.

The design of the architecture was inspired by the Lung Sound Resnet with Non-Local Layer (LungRN + NL) [15]. The spectrogram (STFT) and scalogram (WT) are the input images of the network. Two channels with the same architecture were adopted to study them separately. In each channel, a 2D 1×1 convolution filter was used to expand the channel size of the input image to 64, and the channel size was kept to be the same all the time. The channel size was not expanded because the total amount of ICBHI 2017 data is not large. Therefore, a smaller network could prevent overfitting. However, a smaller network does not mean the network is simple. Instead, the Blnet includes various operations in each block. The Block-I, Block-II, and Self-attention block were applied to study the deep

features of the input image. The details of Block-I and BlocK-II are shown in Figure 3. In both Block-I and Block-II, we integrated the strength of the GoogleNet [18] and ResNet [17]. Convolutions with different kernel sizes and average pooling were adopted to study the features across the time and frequency domain. In both the spectrogram and scalogram, the energy shifts over the time and frequency domain. Significant features could be hidden in a small interval or a larger interval. Thus, three convolution kernels, 3×3, 4×4, and 5×5, were applied to ensure that those features could be captured. In addition, a 3×3 average pooling was adapted to store the average energy at a local place. We used the Relu [24] as our activation function and the group normalization [25] to standardize data. We found the group normalization worked better than batch normalization and instance normalization in Blnet. The image size was reduced by one-half by Block-I, as the stride was equal to two, and a one-by-one convolution layer in Block-I aims to keep the residual information [17] of the input.

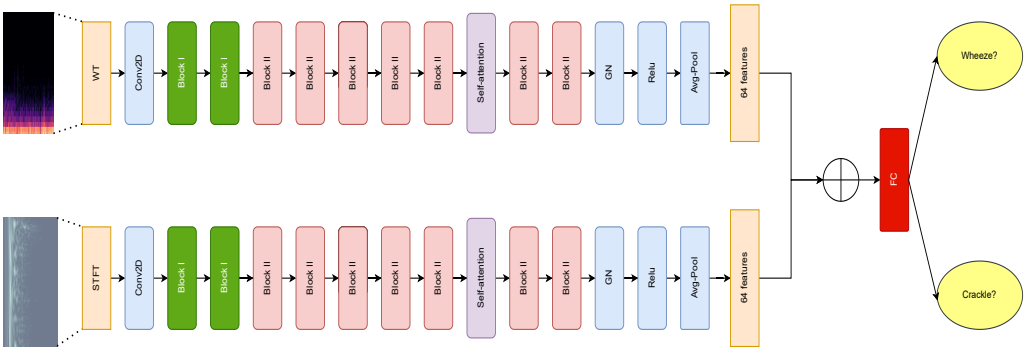


Figure 2. The architecture of the Blnet without the blocking variable in the first stage training.

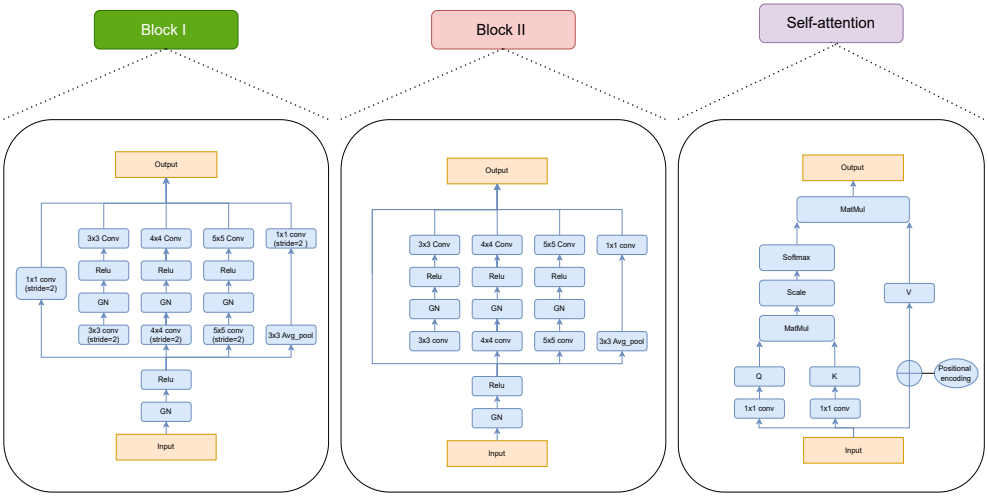


Figure 3. The Block-I and Block-II and Self-attention Block.

A Self-attention block [19] was put in the middle of the Blnet to study the connection between the channels across the time and frequency domain. This operation allows the interaction between the channels. The formula of the attention mechanism is summarized below:

$$Attention(Q, K, V) = softmax(\frac{Q^T K}{\sqrt{d}})V, \quad (2)$$

where V is the input of the self-attention layer, and K and Q are two arrays with the same shape as V, generated by two 1×1 convolution layers, respectively, and d is equal to the channel size, which is 64. To be specific, the image of each channel is firstly stretched into a vector by transferring it from 2D to 1D before applying the self-attention mechanism. A positional encoding is added to each vector in V to retain the positional information

of the image. With the attention matrix, the channels considered to be 'important' by the back-propagation will remain and vice versa. The self-attention mechanism allows global interaction between the channels, which improves the horizon of the convolution neural network. The details of the Self-attention block are shown in Figure 3. Finally, we applied the global average pooling at the end of our encoder. A total of 64 features were extracted from the input image of the STFT and WT, respectively, and sent to the classifier. The fully connected layer was chosen to be our classifier, which performed the same as two independent binary logistic regression models used for classifying the presence of crackles and wheezes within the respiratory cycle.

3.2. Two Stage Training with the Blocking Variable

A two-stage training process with the blocking variable was proposed to improve the performance of the model. Firstly, the network was trained with only the spectrogram and scalogram as the input. However, the distributions of respiratory sound data collected from different chest locations are highly skewed. For example, there is a strong presence of heart-beat sounds in the data collected from the anterior left (Al) and the trachea (Tc) region. Thus, we decided to add the location information at the second stage of training. We turned the blocking variable into one-hot encoding, and each encoding represented the chest location "Tc", "Al", "Ar", "Pl", "Pr", "Ll", and"Lr", respectively. The chest location of the respiratory sound, where it was collected from, would be used as the blocking variable and sent to the classifier together with features learned by the encoder. The details are shown in Figure 4. In the second stage, we retrained the encoder and the classifier, but the encoder was initialized from the first stage of training. We retrained the encoder in the second stage because we wanted our encoder to be slightly modified to fit the new classifier. The benefits of adding the blocking variable to the deep neural network are the same as the benefits of the blocking variable used in the statistical theory of the design of the experiments. Although the blocking variable itself has no effect in classifying the adventitious sound, it reduces the variance of the model error. This is according to the effectiveness of the blocking variable in the experiment design. The blocking variable is effective because we usually assume that all the data are independent and identically distributed. However, the sound collected in different chest positions did not have the same distribution, which would violate the model assumption. Adding the blocking variable lets the network know that data collected from the same chest location belong to a specific cluster, helping back-propagation to extract more robust and efficient features, and produce better prediction results.

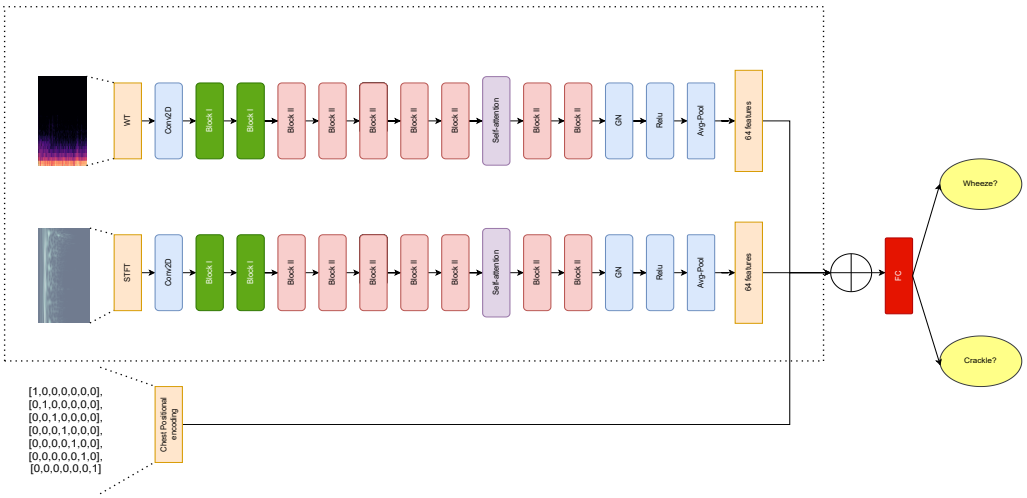


Figure 4. The architecture of the Blnet in the second stage training with the blocking variable. The parameters in the dotted line are initialized from the pre-trained network in the first stage.

3.3. The Loss Function

Since our proposed network has only two outputs, the loss function, shown in Equation (3), is calculated by summation of the binary cross-entropy loss for correctly classifying the existence of wheezes and crackles, respectively.

$$Loss = \sum_{i=1}^{n} \sum_{j=1}^{2} -(W_j y_{ij} \log(\sigma(X_i)_j) + (1 - y_{ij}) \log(1 - \sigma(X_i)_j)), \quad (3)$$

where y_{ij} indicates the label of the data. In specific, i represents the index of the data, $y_{i,j=0}$ represents the label for Crackle, $y_{i,j=1}$ represents the label for Wheeze. $\sigma(X_i)$ is the output of our network, and the hyper-parameter W_j is a weight to adjust the imbalance problem between normal and adventitious data. Further, $W_{j=0} = 0.9$ and $W_{j=0} = 0.95$ is found to work the best. In our experiment, Equation (3) was found to be a better loss function than using cross-entropy loss for a four-class classification task, which makes the training more robust. This is because we have simplified a four-class classification task to two independent binary classification tasks, but in the general case three is required. Correspondingly, the one-hot encoding of the labels of the data has to be slightly modified. In the standard setting, $[1, 0, 0, 0]$, $[0, 1, 0, 0]$, $[0, 0, 1, 0]$, and $[0, 0, 0, 1]$ are used to represent labels of the Normal, Crackles, Wheeze, and Both classes, respectively. However, we suggested using $[0, 0]$, $[1, 0]$, $[0, 1]$, and $[1, 1]$ instead.

4. Experiment Result

4.1. The Evaluation Method and Implement Device

The official ICBHI 2017 evaluation method [8] is used to evaluate the performance of our model. The scoring method is defined to be the average of the sensitivity S_e and specificity S_p in the following.

$$\begin{aligned} S_e &= \frac{P_C + P_W + P_B}{Crackle + Wheeze + Both}, \\ S_p &= \frac{P_N}{Normal}, \end{aligned} \quad (4)$$

where P_C, P_W, P_B, and P_N are the total number of Crackle, Wheeze, Both, and Normal cycles for the correction prediction, respectively. *Crackle*, *Wheeze*, *Both*, and *Normal* are the total number of samples for four classes. To conduct the experiment, we implemented the model by PyTorch in python with a 64-bit Windows machine with Intel i9-10900k 3.50 GHz CPU and NIVIDA RTX2080 GPU. The Adam optimizer was used with an initial learning rate equal to 0.0004 and dropping to one-tenth every 80 epochs. We also used a 15% dropout rate for the encoder and a 25% dropout rate for the classifier to prevent overfitting, and found the optimal batch size is 32.

4.2. Accuracy and Confusion Matrix

We compared our results with the state-of-the-art model result on both the official train–test split and the 8:2 train–test split. The experiment results are shown in Table 2. For the 8:2 train–test split, our final proposed model achieved a 72.72% for the average score and 66.31% for sensitivity, improving the state-of-the-art average score by 4.22%. The model also raised the state-of-the-art sensitivity by 12.61%, which is a massive improvement because the sensitivity is considered a more challenging task than the specificity due to the data imbalance problem. For the official train–test split, our model achieved 51.98% average score, which is the state-of-the-art result if the model is not pre-trained on any other data set. Our model achieved 42.63% sensitivity on the official train–test split, which improves the state-of-the-art result by 2.53%. The details of the confusion matrix are shown in Figure 5. The low sensitivity in the official train–test split can primarily be attributed to the model's inability to accurately identify Wheeze data. This is due to the scarcity of training Wheeze data available within the official train–test split. There are only 491 Wheeze data and

277 Both data in the train set. However, the results are much better in the 8:2 train–test split because of more sufficient data.

Table 2. Comparison of our model with other methods (S_e indicates the sensitivity, and S_p indicates the specificity).

	Method	S_e	S_p	Score
Official Split	Jakovljevic et al. [9]	-	-	39.56 %
	Chambres et al. [10]	20.81%	78.05%	49.43%
	Serbes et al. [11]	-	-	49.86%
	Ma et al. [14]	31.12%	69.20%	50.16%
	Ma et al. [15].	41.32%	63.20%	52.26%
	Gairola et al. [16] (Pre-trained)	40.1%	72.3%	56.2%
	Proposed model stage 1	42.88%	60.05%	51.47%
	Proposed model stage 2 with blocking	42.63%	61.33%	51.98%
8:2 Split	Kochetov et al. [12]	58.43%	73.00%	65.70%
	Acharya et al. [13]	48.63%	84.14%	66.38%
	Ma et al. [15]	63.69%	64.73%	64.21 %
	Gairola et al. [16] (Pre-trained)	53.7%	83.3%	68.5 %
	Our proposed model stage 1	63.99%	78.72%	71.35 %
	Our proposed model stage 2 with blocking	66.31%	79.13%	72.72 %

Table 2 shows that the accuracy improves from 71.35% to 72.72% using two-stage training on the 8:2 train–test split. It proves that adding the blocking variable in a deep learning model improves the model's performance. The effectiveness of the blocking variable grows with the size of the data, as the improvement of the model in the second stage is more significant in the 8:2 train–test split than in the official train–test split. However, we found that the blocking variable was only effective when the distributions between the blocks were heavily skewed. If it is not the case, adding the blocking variable in the fully connected layer will introduce noise, which may even lead to an opposite effect. We also conducted the experiment by adding other possible blocking variables, e.g., Acquisition mode and Recording equipment. However, no significant improvement in the prediction result could be found. It is worth mentioning that different recording equipment has different sampling frequencies, which do have an effect on the distribution of the input picture. However, this problem has already been addressed in the pre-processing stage.

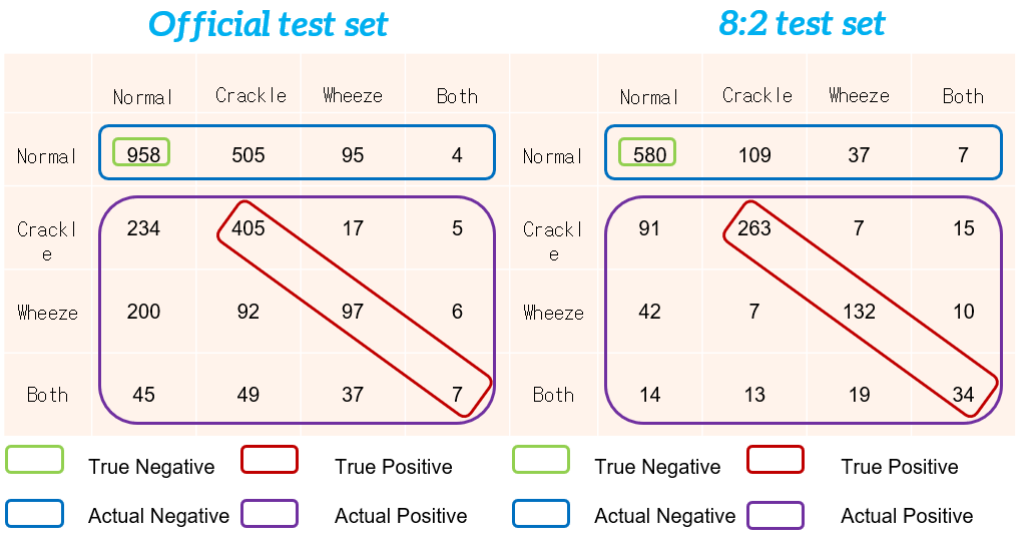


Figure 5. Confusion matrices for two splits of the data.

Other techniques that were found to be effective can be summarized as follows. We found that adopting the different kernel sizes and average pooling increased the accuracy of the model in the first stage of training by 1.21% in the 8:2 train–test split. To tackle the problem of the different sampling frequencies between devices, we let the hop length be proportional to the sampling frequency and kept the window size the same, which improved the accuracy by 0.83%, compared to the result where both the hop length and the window length were proportional to the sampling frequency. The two-grain network with both the STFT and WT as input improved the accuracy by 1.68% compared to the single-grain network taking only STFT as an input. The Self-attention block was found to be the most effective when placed in front of the second-last Block-II; the model's accuracy is reduced by 2.04% if it is removed. The main difference between Block-I and Block-II is that Block-I reduces the size of the input image by a half and Block II does not. The accuracy of the model is reduced by 0.72% and 4.29% when two and four Block-II are replaced by Block-I, respectively. The two-output loss function we proposed also surpasses the standard four-output loss function to handle the four classes classification task, as evidenced by a 1.44% improvement in accuracy. All the research of our methods was conducted on the 8:2 train–test split because it produced more stable results than the official train–test split. The details of the ablation test results are shown in Table 3.

Table 3. Ablation test results (S_e indicates the sensitivity, and S_p indicates the specificity).

Method	S_e	S_p	Score
Our proposed method	63.99%	78.72%	71.35%
Without fixed window size	61.10%	79.94%	70.52%
Without multiple kernels	62.19%	78.09%	70.14%
Without wavelet transform	58.21%	81.13%	69.67%
Without Self-attention	59.01%	79.61%	69.31%
Four-output loss function	59.51%	80.31%	69.91%
Four Block-I layers	59.75%	81.51%	70.63%
Six Block-I layers	55.52%	78.59%	67.06%

4.3. Discussion

Firstly, we found the short-time Fourier transform and wavelet transform were the most effective inputs of the convolution neural network model. The short-time Fourier transform takes out the features in the time domain and frequency domain, and the wavelet transform balances the local spectral and temporal information that provides the additional information that short-time Fourier transform does not capture. Secondly, our Blnet network outweighed all the other convolution neural networks in the respiratory sound classification task, evidenced by producing the best prediction result. This is mainly attributed to the fact that the Blnet network incorporates the strength of the ResNet, GoogleNet, and the self-attention layer, and that we used a slightly smaller deep neural network to adapt to the limited data problem. In common architectures of the convolution neural network, people expand the channel size and reduce the size of the image, but we found that continuing to expand the channel size and reducing the size of the image did not make the performance of the model better due to the insufficient data. Thus, we kept the channel size to be 64 and only reduced the size of the image from 128 to 32 through 2 Block-I layers. The Blnet network has 1,147,074 parameters in total. By comparison, even the smallest ResNet-16 has 14,356,544 parameters. Thirdly, using different kernel sizes and average pooling is advantageous because the adventitious sound could be between a small time interval and a large time interval. Furthermore, adding the self-attention layer is an effective way to improve the model's performance. This is because one dimension of the spectrogram and scalogram is time, and there will be connections between the pixels in that dimension. The attention mechanism successfully expands the horizon of the network and helps the network retain more robust information and remove less important information. Fourthly, a simplified loss function is suggested in this paper,

which facilitates the network's understanding of the relationship between those four classes. Finally, including the blocking variable in the neural network was proven to be an effective way to improve the performance of a deep neural network. Other researchers can easily add the blocking variable into their framework because the only adjustment made is on the fully connected layer. A two-stage training process is proposed instead of training with the blocking variable from the beginning. This is because the blocking variable is only effective when the model has already extracted robust features. Although the blocking variable itself is not helpful for the classification, it could influence the features extracted by the model and their impact on the classification. If it is put into the model at the beginning, it would be more likely to be considered noise by the back-propagation. This is because the model will only consider whether the information itself is useful at the beginning, yet each cluster in Table 1 has the similar proportion of four labels.

5. Conclusions and Future Work

We proposed the Blnet network to implement the automatic classification task of the ICBHI 2017 respiratory sound data, along with a two-stage training process with the blocking variable and several useful techniques. Those useful techniques include performing the data augmentation within the clusters, adding the attention layer, choosing a simplified loss function, adopting multiple kernel sizes and average pooling, and using the specific sliding window of the STFT. These methods can be easily incorporated into other research frameworks. In the future, the usage of the blocking variables could be further investigated. It could be used not only on respiratory sound data, but also on any other deep learning framework, as long as there is a non-IID data problem in the study data set. There are a lot of medical data that may face the same problem. The state-of-the-art result was achieved in the four-class classification task based on the ICBHI 2017 scoring standard. Our proposed model scored at the 72.72% on the 8:2 train–test split, and 51.98% on the official train–test split. The result of the official train–test split is restricted mainly by the lack of sufficient training data, especially the Wheeze data. The result can be greatly improved if the model is pre-trained on any other new collected respiratory sound data in the future.

Author Contributions: Methodology, R.Y. and M.S.; formal analysis, R.Y. and Y.H.; investigation, R.Y. and Y.H.; writing—original draft preparation, R.Y.; writing—review and editing, K.L., J.Y., J.L. and R.Y.; supervision, M.S., J.Y. and J.L.; project administration, R.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by "Medtronic Technology Center, Shanghai" and the grant number of the FUNDER is 19H100000177.

Data Availability Statement: The project is conducted based on the largest open respiratory sound database, and the link to the data could be found at https://bhichallenge.med.auth.gr/ (accessed on 22 March 2019).

Acknowledgments: We would like to express our sincere gratitude to Yingying Liu, Yi Wu, Zhenhua Yue, and Pengjia Cao at the Shanghai Medtronic Technology Center, who worked as a team with us for ten months, for providing their kindly support, suggestions, and supervision of this project.

Conflicts of Interest: The author Sun Mingxia, who works for the Shanghai Medtronic Technology Center (FUNDER), participated in the design of the study and her suggestions regarding the methodology were accepted. Despite that, all authors declare no other conflicts of interest.

References

1. Mangione, S.; Nieman, L.Z. Pulmonary auscultatory skills during training in internal medicine and family practice. *Am. J. Respir. Crit. Care Med.* **1999**, *159*, 1119–1124. [CrossRef] [PubMed]
2. Xie, J.; Tong, Z.; Guan, X.; Du, B.; Qiu, H.; Slutsky, A.S. Critical care crisis and some recommendations during the COVID-19 epidemic in China. *Intensive Care Med.* **2020**, *46*, 837–840. [CrossRef] [PubMed]

3. Adeloye, D.; Elneima, O.; Daines, L.; Poinasamy, K.; Quint, J.K.; Walker, S.; Brightling, C.E.; Siddiqui, S.; Hurst, J.R.; Chalmers, J.D.; et al. The long-term sequelae of COVID-19: An international consensus on research priorities for patients with pre-existing and new-onset airways disease. *Lancet Respir. Med.* **2021**, *9*, 1467–1478. [CrossRef] [PubMed]
4. Nagasaka, Y. Lung sounds in bronchial asthma. *Allergol. Int.* **2012**, *61*, 353–363. [CrossRef] [PubMed]
5. Weiss, E.B.; Carlson, C.J. Recording of breath sounds. *Am. Rev. Respir. Dis.* **1972**, *105*, 835–839. [PubMed]
6. Vyshedskiy, A.; Alhashem, R.M.; Paciej, R.; Ebril, M.; Rudman, I.; Fredberg, J.J.; Murphy, R. Mechanism of inspiratory and expiratory crackles. *Chest* **2009**, *135*, 156–164. [CrossRef] [PubMed]
7. Munakata, M.; Ukita, H.; Doi, I.; Ohtsuka, Y.; Masaki, Y.; Homma, Y.; Kawakami, Y. Spectral and waveform characteristics of fine and coarse crackles. *Thorax* **1991**, *46*, 651–657. [CrossRef] [PubMed]
8. Rocha, B.M.; Filos, D.; Mendes, L.; Serbes, G.; Ulukaya, S.; Kahya, Y.P.; Jakovljevic, N.; Turukalo, T.L.; Vogiatzis, I.M.; Perantoni, E.; et al. An open access database for the evaluation of respiratory sound classification algorithms. *Physiol. Meas.* **2019**, *40*, 035001. [CrossRef] [PubMed]
9. Jakovljević, N.; Lončar-Turukalo, T. Hidden markov model based respiratory sound classification. In *Precision Medicine Powered by pHealth and Connected Health, Proceedings of the ICBHI 2017, Thessaloniki, Greece, 18–21 November 2017*; Springer: Singapore, 2017; pp. 39–43.
10. Chambres, G.; Hanna, P.; Desainte-Catherine, M. Automatic detection of patient with respiratory diseases using lung sound analysis. In Proceedings of the 2018 International Conference on Content-Based Multimedia Indexing (CBMI), La Rochelle, France, 4–6 September 2018; pp. 1–6.
11. Serbes, G.; Ulukaya, S.; Kahya, Y.P. An automated lung sound preprocessing and classification system based onspectral analysis methods. In *Precision Medicine Powered by pHealth and Connected Health, Proceedings of the ICBHI 2017, Thessaloniki, Greece, 18–21 November 2017*; Springer: Singapore, 2017; pp. 45–49.
12. Kochetov, K.; Putin, E.; Balashov, M.; Filchenkov, A.; Shalyto, A. Noise masking recurrent neural network for respiratory sound classification. In *Artificial Neural Networks and Machine Learning—ICANN 2018, Proceedings of the 27th International Conference on Artificial Neural Networks, Rhodes, Greece, 4–7 October 2018*; Springer: Singapore, 2018; pp. 208–217.
13. Acharya, J.; Basu, A. Deep neural network for respiratory sound classification in wearable devices enabled by patient specific model tuning. *IEEE Trans. Biomed. Circuits Syst.* **2020**, *14*, 535–544. [CrossRef] [PubMed]
14. Ma, Y.; Xu, X.; Yu, Q.; Zhang, Y.; Li, Y.; Zhao, J.; Wang, G. LungBRN: A smart digital stethoscope for detecting respiratory disease using bi-resnet deep learning algorithm. In Proceedings of the 2019 IEEE Biomedical Circuits and Systems Conference (BioCAS), Nara, Japan, 17–19 October 2019; pp. 1–4.
15. Ma, Y.; Xu, X.; Li, Y. LungRN+ NL: An Improved Adventitious Lung Sound Classification Using Non-Local Block ResNet Neural Network with Mixup Data Augmentation. In Proceedings of the Interspeech, Shanghai, China, 25–29 October 2020; pp. 2902–2906.
16. Gairola, S.; Tom, F.; Kwatra, N.; Jain, M. Respirenet: A deep neural network for accurately detecting abnormal lung sounds in limited data setting. In Proceedings of the 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Virtual, 1–5 November 2021; pp. 527–530.
17. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
18. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 12 June 2015; pp. 1–9.
19. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*.
20. Bohadana, A.; Izbicki, G.; Kraman, S.S. Fundamentals of lung auscultation. *N. Engl. J. Med.* **2014**, *370*, 744–751. [CrossRef] [PubMed]
21. Bracewell, R.N.; Bracewell, R.N. *The Fourier Transform and Its Applications*; McGraw-Hill: New York, NY, USA, 1986; Volume 31999.
22. Bentley, P.M.; McDonnell, J. Wavelet transforms: An introduction. *Electron. Commun. Eng. J.* **1994**, *6*, 175–186. [CrossRef]
23. Zhang, H.; Cisse, M.; Dauphin, Y.; Lopez-Paz, D. mixup: Beyond empirical risk management. In Proceedings of the 6th International Conference Learning Representations (ICLR), Vancouver, BC, Canada, 30 April 30–3 May 2018; pp. 1–13.
24. Glorot, X.; Bordes, A.; Bengio, Y. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*; JMLR Workshop and Conference Proceedings; Mlr Press: Mishawaka, IN, USA, 2011; pp. 315–323.
25. Wu, Y.; He, K. Group normalization. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.