INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Coughing-based Recognition of Covid-19 with Spatial Attentive ConvLSTM Recurrent Neural Networks

Tianhao Yan[1,2]*, Hao Meng*[1]*, Emilia Parada-Cabaleiro*[3]*, Shuo Liu*[2]*, Meishu Song*[2]*, Björn W. Schuller*[2,4]

[1]College of Intelligent Systems Science and Engineering, Harbin Engineering University, Harbin, 150001, China
[2]Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Germany
[3]Institute of Computational Perception, Johannes Kepler University Linz, Austria
[4]GLAM – Group on Language, Audio, & Music, Imperial College London, UK

menghao@hrbeu.edu.cn

Abstract

The rapid emergence of COVID-19 has become a major public health threat around the world. Although early detection is crucial to reduce its spread, the existing diagnostic methods are still insufficient in bringing the pandemic under control. Thus, more sophisticated systems, able to easily identify the infection from a larger variety of symptoms, such as cough, are urgently needed. Deep learning models can indeed convey numerous signal features relevant to fight against the disease; yet, the performance of state-of-the-art approaches is still severely restricted by the feature information loss typically due to the high number of layers. To mitigate this phenomenon, identifying the most relevant feature areas by drawing into attention mechanisms becomes essential. In this paper, we introduce Spatial Attentive ConvLSTM-RNN (SACRNN), a novel algorithm that is using Convolutional Long-Short Term Memory Recurrent Neural Networks with embedded attention that has the ability to identify the most valuable features. The promising results achieved by the fusion between the proposed model and a conventional Attentive Convolutional Recurrent Neural Network, on the automatic recognition of COVID-19 coughing (73.2 % of Unweighted Average Recall) show the great potential of the presented approach in developing efficient solutions to defeat the pandemic.
Index Terms: SARS-CoV-2 detection, Computer Audition, Spatial Attention, CNN, RNN, LSTM, Spectrogram, ACRNN.

1. Introduction

Since 2019, human beings are threatened by the COVID-19 pandemic, declared by the World Health Organization as a global public health crisis [1] [2]. The dramatic death rate, devastated economy, and individuals' freedom restrictions derived by the ongoing pandemic, continue to have an extreme negative impact on human life [3] [4]. To overcome this critical situation, one of the main difficulties that need to be addressed is the still insufficient detection of the disease, something essential to control its spread. To this end, developing mechanisms that identify infected individuals at an early stage is crucial [5] [6]. In order to cope with this medical challenge, the astonishing achievements of deep learning (DL) algorithms has expeditiously turned them into powerful tools for medical diagnosis [7], showing to be very promising in the COVID-19 detection, too [8].

State-of-the-art research aimed to identify COVID-19 through DL methods is typically based on convolutional neural networks (CNN), especially when considering medical images, such as chest X-ray, as input data [9]. Nevertheless, the performance of CNN is typically restricted by the feature information loss characteristic of very deep architectures. In this regard, recurrent neural networks (RNN), such as long-short term memory (LSTM) RNN—particularly suited for classifying continuous data due to their capability to remember characteristics of the signal displayed earlier—have been presented in order to alleviate the degradation problem [10]. Yet, to mitigate the information loss, identifying the most relevant feature areas by drawing into attention mechanisms is also essential, something especially promoted by the breakthrough of Transformers in natural language processing (NLP), whose attention mechanism enables to efficiently identify the most relevant feature area of the network. Indeed, this advance has already shown successful results for NLP-based COVID-19 applications [11].

Since early detection is still an open challenge in the fight against the pandemic, efficient methods able to easily detect the infection from a broader variety of symptoms are particularly necessary. In this regards, the development of coughing-based systems for automatic COVID-19 recognition are especially useful in supporting the ongoing DL-based diagnosic tools [8]. To this end, we propose Spatial Attentive ConvLSTM-RNN (in short, SACRNN), a novel deep learning architecture based on convolution recurrent neural networks with embedded attention for the automatic detection of COVID-19 coughing [12]. Our approach employs a CNN to extract the most relevant characteristics of the FFT-based Mel-spectrogram representation of the audio signal. Subsequently, after passing a series of LSTM layers, the feature maps are connected to an Attention structure and to a GlobalAveragePooling Layer. Finally, the model is enhanced by fusing it to a conventional Attentive Convolutional Recurrent Neural Network (ACRNN) [13].

2. Related Work

The acoustic features typically used for speech classification are mainly divided into two categories: 1) the 'traditional' features, which include frame-based Low Level Descriptors (LLDs) and utterance-based High-level Statistical Functionals (HSFs) [14] [15] [16]; 2) the spectrogram-based features [17] [18]. In the past, the traditional features, typically extracted with toolkits such as openSMILE [19], were feed into machine learning algorithms, e. g., Support Vector Machine (SVM), XGBoost, or Random Forests [20] [21] [22]. Nowadays, with the advent of DL, researchers began to successfully feed DL models with raw audio representations [23], such as spectrograms [24].

Inspired and benefiting from the achievements reached in Computer Vision and NLP, most of the speech classification research presented in the recent years is based on DL models,

http://dx.doi.org/10.21437/Interspeech.2021-630

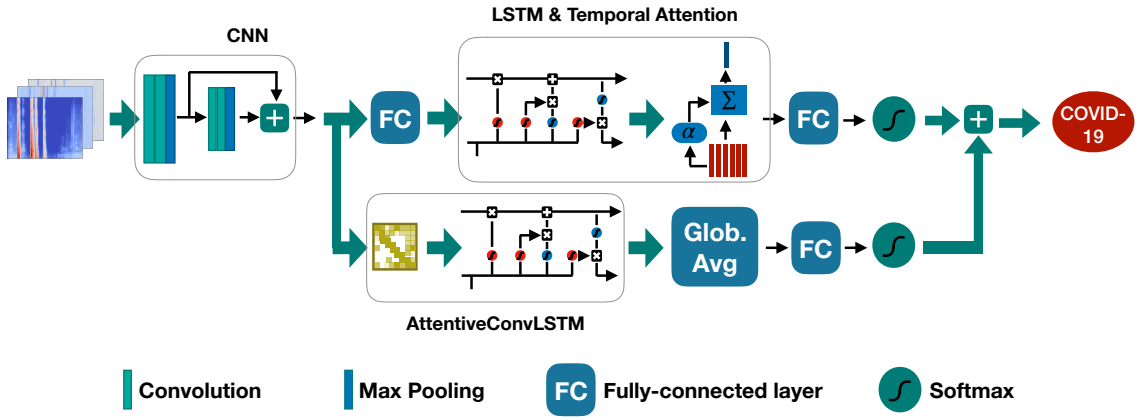

Figure 1: *After the 3-channel log-Mel-spectrogram are input to the CNN, the model consists of two parts: the ACRNN (above) and the SACRNN (below), which jointly employ a set of convolutional layers with ResNet. While SACRNN uses the Spatial AttentiveConvLSTM structure, ACRNN uses LSTM and temporal attention. The two models' features each go through a softmax layer and are fused additively.*

e. g., through the use of CNN, LSTM, and Attention mechanisms [25]. For instance, great success has been achieved by feeding DL models with Mel-spectrograms of the audio signals [26]. In the realm of paralinguistics research, a variety of DL-based approaches have been developed for speech classification [27] [28] [29]. In particular, the novel algorithm Atention-based Convolutional RNN (ACRNN), able to learn discriminative features for Speech Emotion Recognition, has been recently presented [13]. Outstanding performance in distinguishing emotional distribution has also been achieved by the use of frame-level features with attention-based LSTM networks [30].

Although the spectrogram feature maps from audio signals (analogous to the channel in images) present certain time-frequency domain characteristics, from which attentive feature information can be extracted, this has not yet been taken into consideration by any of the already presented approaches. Considering this, and inspired by the outstanding results achieved in computer vision by the use of specific Attentive Convolutional LSTM architectures [31], we add an attentive LSTM structure to the output of a CNN in order to specifically extract more centralised temporal-frequency feature maps. By extracting the features of the LSTM-Attention layer in the time dimension of the spectrograms, these can be connected with an AttentiveConvLSTM layer, so that the model has the ability to locate and re-extract the most relevant features of the audio signal.

3. Model Architecture

To build the proposed architecture (SACRNN), we first apply Voice Activity Detection (VAD) on the audio signal to cut out the silent frames at the beginning and the end of the recordings. Subsequently, we generate three type of Mel-spectrograms: static, deltas, and delta-deltas; which will be used as input of the DL model. The proposed model incorporates a CNN structure on top of a RNN aimed to enhance the effect of the attention mechanism. In specific, the spatial block framework, followed by a global-average-pooling layer and a softmax layer, is fused with a conventional ACRNN framework. Note that the two branch models share the same convolutional layer, as shown in Figure 1.

3.1. Mel-Spectrogram Generation

In order to focus on the coughing signal while disregarding static frames, VAD is applied through the Python data package `webrtcvad`. Following [32], who showed that 2-dimensional convolutions are more efficient than the 1-dimensional ones in small datasets, 2-dimensional Mel-spectrograms are generated through Fast Fourier Transform (FFT) from the audio signal. In order to better capture the dynamic information of the signal, deltas and delta-deltas are also extracted from the Mel-spectrograms. The samples are set to a sample rate of 16 kHz and the Mel-spectrograms are generated considering 40 filterbanks and a Hamming window of 25 ms with a shift of 10 ms. The energy spectrum is computed by passing the Short-Time Fourier Transform (STFT) of the raw signal through the Mel-filterbanks, by this obtaining the Mel-spectrogram M. Subsequently, the logarithm of the generated Mel-spectrogram is performed, as shown in Equation (1).

$$M\left(i\right) = \sum_{k=f(i-1)}^{f(i+1)} log\left(H_i\left(k\right)\right) * \left|X\left(k\right)\right|^2, \qquad (1)$$

where $\left|\mathbf{X}\left(\mathbf{k}\right)\right|^{\mathbf{2}}$ represents the energy spectrum, $\mathbf{H_i}\left(\mathbf{k}\right)$ indicates the Mel-filterbanks, $\mathbf{k}$ is the output of the FFTs, and $\mathbf{i}$ is the number of filterbanks. Subsequently, from the 1 channel static Log-Mel Spectrogram, deltas and double-deltas are calculated as shown in Equation (2) and Equation (3) respectively. Thus, leading to the 3-channel Spectrogram structure.

$$M\left(i\right)^d = \frac{\sum_{n=1}^{N} n\left(M\left(i\right)_{t+n} - M\left(i\right)_{t-n}\right)}{2\sum_{n=1}^{N} n^2} \qquad (2)$$

$$M\left(i\right)^{dd} = \frac{\sum_{n=1}^{N} n\left(M\left(i\right)_{t+n}^{d} - M\left(i\right)_{t-n}^{d}\right)}{2\sum_{n=1}^{N} n^2}, \qquad (3)$$

where $\mathbf{t}$ is the number of frames, $\mathbf{d}$ indicate the number of delta, $\mathbf{n}$ represents the number of differences between the current frame and the previous-next frames, and $\mathbf{N}$ is set to 2 under normal situation. Hence, the output dimension of the 3-channel spectrogram, which represents the values of static, deltas, and delta-deltas, respectively (input for the CNN), is $\mathbf{M} \in \mathbf{R}^{\mathbf{t},\mathbf{f},\mathbf{c}}$, where $\mathbf{t}$ indicates the time, $\mathbf{f}$ is the number of filters, and $\mathbf{c}$ is the number of channels.

3.2. Model

From the image domain perspective, adding Attention layers has as a goal to focus on specific image target areas. When

Table 1: *Parameters of the used SACRNN model*

Parameters	OutputDim	Kernel_Size
Conv. Block	T * N *32	3*3
MaxP. Layer	T/2 * N/2 *32	2*2
Convo Block	T/2 * N/2 *64	3*3
MaxP. Layer	T/2 * N/4 *64	1*2
Reshape	T_S * T/2 * N/4 *128	-
• AttConvLSTM	T/2 * N/4 *128	3*3
Global.Avg	128	-
Dense (Dropout)	(rate = 0.2)64	-
Softmax	**K**	-

transferring this idea to the audio domain, finding the target area requires more effort. To do so, beyond adding the time dimension to the 2-dimensional feature space, we also add an LSTM layer to the output of the CNN layer, by this enhancing the attention effect in achieving the target area. Therefore, we analyse the Mel-spectrogram from two perspectives: the strengthening feature maps and the temporal domain information, corresponding to SACRNN and ACRNN, respectively.

3.2.1. CNN Model

The 3-channel 2-dimensional Mel-Spectrograms (cf. Section 3.1) were used as input for a CNN with 2 convolutional blocks. Each block contains two identical convolutional layers which include the same kernels and sizes (cf. Table 1). First, we connect a maxpooling layer with a pooling size of 2*2 after the first convolutional block. Different from the second maxpooling layer, the size is equals 1*2. In order to preserve the original features to the maximum, similar to the ResNet structure, we make a skip connection between the output of the first and second layers of convolution.

3.2.2. SACRNN model

The outputs from the CNN are sent to the SACRNN and the ACRNN model. Inspired by [31], we add a time axis to the output dimension of the convolutional layer which constitutes a 3-dimensional input tensor. Thus, the traditional LSTM is extended to a spatial model. By this, we exploit the excellent performance of LSTM in processing continuous feature maps attractively while enhancing at the same time the performance of the attention mechanism, which enables to capture more effective features in the channel dimension. Note that we use the 2-dimensional spectrogram as input to the AttentiveConvLSTM layer. In Table 2, the model parameters are shown, where $\mathbf{K}$ is 2, since we are dealing with a binary classification problem i. e., either Positive or Negative (COVID-19) sample.

By replacing the dot product operation of the LSTM with the convolution in the calculation, we can make use of the LSTM layer to better retain the characteristics. First, we combine the CNN output with the time axis (iteration axis) by adding the hidden state $\mathbf{H_{t-1}}$ to the Tanh activation function, which is performed by a convolutional layer with a kernel of 1 named $\mathbf{R_a}$, as shown in Equation (4).

$$Z_t = R_a * tanh(W_a * X + *H_{t-1} + b_a), \tag{4}$$

where, $\mathbf{W_a}$ and $\mathbf{U_a}$ represent the convolutional kernel of $\mathbf{X}$ and $\mathbf{H_{t-1}}$ respectively. Subsequently, the attention mechanism is carried out by applying Softmax (in the spatial dimension) to the 2-dimensional tensor, as shown in Equation (5).

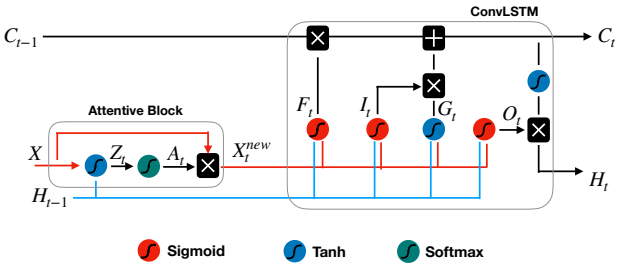

Figure 2: *Structure of the Spatial AttentiveConvLSTM network. X is the input into the network based on 2-dimensional maps.*

$$A_t^{ij} = p\left(X, H_{t-1}\right) = \frac{exp\left(R_t^{ij}\right)}{\sum\sum_j^i exp\left(R_t^{ij}\right)}, \tag{5}$$

where $\mathbf{A_t^{ij}}$ represents the $(\mathbf{i}, \mathbf{j})$ position of the element in the tensor. Thus, in order to obtain the new input $\mathbf{X^{new}}$, the attention map is exploited to update the input $\mathbf{X}$ with an element-wise product between feature maps and the attention map in each channel as shown in Equation (6).

$$X_t^{new} = A_t \odot X. \tag{6}$$

After applying the attention mechanism, the new input goes to the LSTM layer by updating each parameter as shown in Equation (6) - (12).

$$I_t = \sigma\left(W_i * X_t^{new} + U_i * H_{t-1} + b_i\right) \tag{7}$$

$$F_t = \sigma\left(W_f * X_t^{new} + U_f * H_{t-1} + b_f\right) \tag{8}$$

$$O_t = \sigma\left(W_o * X_t^{new} + U_o * H_{t-1} + b_o\right) \tag{9}$$

$$G_t = tanh\left(W_c * X_t^{new} + U_c * H_{t-1} + b_c\right) \tag{10}$$

$$C_t = F_t \odot C_{t-1} + I_t \odot G_t \tag{11}$$

$$H_t = O_t \odot tanh\left(C_t\right), \tag{12}$$

where $\mathbf{I_t}$, $\mathbf{F_t}$, $\mathbf{O_t}$ are the three gates respectively, $\mathbf{G_t}$ belongs to link memory, $\mathbf{C_{t-1}}$ and $\mathbf{C_t}$ are both memory cells, $\mathbf{H_{t-1}}$, and $\mathbf{H_t}$ are both hidden states. All of the above are 3D tensors with 256 channels in the spatial LSTM, while $\boldsymbol{W}$ and $\boldsymbol{U}$ are both 2-dimensional convolution kernels. Note that all ***bias*** can be learnable. The whole structure is shown as Figure 2. Finally, the output of the attentiveConvLSTM layer goes into a globalAverage2D layer and a softmax layer.

3.2.3. ACRNN Model

Since finding the area of attention in audio data is a continuous process, it is necessary to connect an LSTM layer in order to retain the continuous features. Subsequently, an attention Layer should also be connected to extract more effective features in the time dimension, as shown in the ACRNN framework. In Table 2, the parameters of the ACRNN are given. Note that the Conv.Blocks from Table 2 have already included maxpooling operation. As for the ACRNN, the input of the model is again the output of the convolutional layer. As another branch of the output of the convolutional layer, we reshape it into a shape of $[\mathrm{batch\_size}, \mathrm{time\_step}, \mathrm{features}]$, where features are the multiplication between n_mel and channel, and the 'time' dimension is retained as the 'timestep' parameter. In order to reduce the dimensionality, following [33], a linear layer with 256 output units is added between the convolutional layer and the LSTM layer containing 128 cells for temporal summarisation. In addition to this, in order to obtain the high-level features representation,

Table 2: *Parameters of the ACRNN model*

Parameters	OutputDim	Kernel_Size
Conv. Block	T/2 * N/4 *64	3*3
Reshape	T/2 * [N/4 *128]	-
Dense	T/2 * 256	256
LSTM	T/2 * 128	128
Attention	T/2 * 128	128
Dense (Dropout)	(rate = 0.2)64	-
Softmax	**K**	-

after employing the LSTM Layer, another round of attention mechanism is performed for the time step. This is preferred, instead of simply performing a global average pooling over time, in order to score the importance of a series of high-level representations. Finally, the output from the above structures is sent to a 64-unit fully connected layer, by this gaining a high-level representation.

4. Experiment

4.1. Dataset Description

For our experiments, a subset of the Cambridge COVID-19 Sound database [34], as used in the COVID-19 Cough Sub-Challenge (CCS) from the INTERSPEECH 2021 Computational Paralinguistics ChallengE (ComParE) [12], is considered. The data, collected in a crowdsourcing manner through the COVID-19 Sounds App, has a length of 1.63 hours and encompasses a total of 929 samples produced by 397 participants (each producing at least 3 samples). To collect the data, participants with both, positive and negative COVID-19 tests, were requested to provide forced coughs.

4.2. Experimental Setup

For the experiments, the partitioning proposed in the COVID-19 CCS is followed as given in the Challenge for experiments, where the sets are 286 samples for training, 231 for validation, and 208 for testing. As evaluation metrics, since the distribution between the two classes in the test set is imbalanced, both the accuracy, i. e., the Weighted Average Recall, as well as the Unweighted Average Recall (UAR) are considered. In addition, as common practice in the medical-related research, the results on the test set will also be reported in terms of specificity, sensitivity, and confidence intervals. Due to the small size of the dataset, in order to optimise parallel acceleration, after extracting the spectrograms, the coughing samples were split into sub-segments of 300 frames. For sub-segments shorter than 300 frames and longer than 100, zero padding was applied at the end. Note that the segmentation was performed on all the sets and the training set was fixed for both validation and test phases.

The model is trained with the sub-segments, which in the testing phase are considered to assess whether the prediction coincides with the one of the real label. Then, the proportion of sub-segments successfully predicted (Voting Rule) is computed [13]. If the proportion is greater than 0.5 w. r. t. the whole sample, the prediction is considered as the real label, otherwise it is regarded as a failed prediction. Before training, the coughing data of all sets is normalised by the overall average and variance of the training set. For comparability, along to the outcomes achieved by the proposed architecture, results for the end-to-end learning End2You approach [12], as well as the results from the SACRNN and the ACRNN individually, i. e., without fusion, are also given. To carry out the experiments, the overall architecture was implemented with Keras [35], using the Adam optimiser with Nestotov momentum [36]. Concerning the number of layers, different depth levels were considered to find the optimal configuration.

Table 3: *Overall results for the four evaluated methods. Accuracy (Acc.), Unweighted Average Recall (UAR), Sensitivity (SE), Specificity (SP), and Confidence Intervals (CI), are given (%). CIs are measured based on the UAR.*

Methods	Acc.	UAR	SE	SP	CI
End2You	–	64.70	–	–	–
SACRNN	83.65	58.40	17.95	98.82	12.38
ACRNN	83.17	70.91	51.28	90.53	16.22
Fusion	**83.65**	**73.20**	**56.41**	89.94	15.01

5. Results and Discussion

In Table 3, our results show that the UAR and sensitivity are lower for SACRNN: 58.4 % and 17.95 % respectively, which indicates that the SACRNN alone is not sufficiently sensitive to recognise positive_coughing. However, it can be seen from specificity that the SACRNN is close to 100 % in the classification of negative_coughing, which indicates that our proposed model is valuable for the negative data. We interpret that this is due to the fact that the SACRNN shows excessive strengths concerning the features based on the spatial spectrogram while it ignores the inherent time-domain information, which results in lower recognition for positive_coughing. Indeed, an improvement is shown when employing the ACRNN alone (70.91 % and 51.28 % for UAR and sensitivity), which confirms the importance of the continuous temporal characteristics. In this case, as expected, the recognition rate of negative_coughing drops, as shown by the lower specificity (90.53 %), hence the fusion model taking advantage from both, the SACRNN and the ACRNN, presents the best performance, exceeding the baseline deep learning model (End2You) algorithm by 8.5 % UAR (cf. 73.2 % vs 64.7 % for Fusion vs End2You in Table 3). The results for sensitivity and specificity in the fusion model (56.41 % and 89.94 %, respectively) show the robustness and generalisation ability of the proposed approach, which is particularly efficient even with noisy samples – as those typically collected in-the-wild.

6. Conclusion

In this paper, we proposed a novel approach based on spatial attentive convolutional recurrent neural networks (SACRNN) based on three-channel spectrogram-based features fused with ACRNN structure. The promising results of our approach for COVID-19 coughing detection, acquiring 73.2 % UAR and considerably outperforming the baseline results, shows that it can be successfully applied to detect infected users, by this supporting medical diagnosis at early stage of the disease, something urgently needed in the current global crisis. In addition, the fusion algorithm, which connect our proposed model and the conventional ACRNN, constitutes a novel idea with potential applications in speech classification tasks in general. In the future, one needs to investigate strategies to minimise the influence of the imbalanced sample size in the models' performance, which has particularly affected the presented study. Besides, a larger amount of data should be explored for improving the generalisability and robustness of the proposed method.

7. References

[1] C. Martinez-Perez, C. Alvarez-Peregrina, C. Villa-Collar, and M. Á. Sánchez-Tena, "Citation network analysis of the novel coronavirus disease 2019 (covid-19)," *International Journal of Environmental Research and Public Health*, vol. 17, no. 20, p. 7690, 2020.

[2] J. Han, K. Qian, M. Song, Z. Yang, Z. Ren, S. Liu, J. Liu, H. Zheng, W. Ji, T. Koike *et al.*, "An early study on intelligent analysis of speech under covid-19: Severity, sleep quality, fatigue, and anxiety," *arXiv preprint arXiv:2005.00096*, 2020.

[3] N. Vindegaard and M. E. Benros, "Covid-19 pandemic and mental health consequences: Systematic review of the current evidence," *Brain, behavior, and immunity*, vol. 89, pp. 531–542, 2020.

[4] E. M. Onyema, N. C. Eucheria, F. A. Obafemi, S. Sen, F. G. Atonye, A. Sharma, and A. O. Alsayed, "Impact of coronavirus pandemic on education," *Journal of Education and Practice*, vol. 11, no. 13, pp. 108–121, 2020.

[5] Z. Luo, M. J. Y. Ang, S. Y. Chan, Z. Yi, Y. Y. Goh, S. Yan, J. Tao, K. Liu, X. Li, H. Zhang *et al.*, "Combating the coronavirus pandemic: Early detection, medical treatment, and a concerted effort by the global community," *Research*, vol. 2020, 2020.

[6] Z. Allam, G. Dey, and D. S. Jones, "Artificial intelligence (ai) provided early detection of the coronavirus (covid-19) in china and will influence future urban health policy internationally," *AI*, vol. 1, no. 2, pp. 156–165, 2020.

[7] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[8] J. Nayak, B. Naik, P. Dinesh, K. Vakula, B. K. Rao, W. Ding, and D. Pelusi, "Intelligent system for covid-19 prognosis: a state-of-the-art survey," *Applied Intelligence*, pp. 1–31.

[9] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "Classification of covid-19 in chest x-ray images using detrac deep convolutional neural network," *Applied Intelligence*, vol. 51, no. 2, pp. 854–864, 2021.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Deep learning applications for covid-19," *Journal of Big Data*, vol. 8, no. 1, pp. 1–54, 2021.

[12] B. W. Schuller, A. Batliner, C. Bergler, C. Mascolo, J. Han, I. Lefter, H. Kaya, S. Amiriparian, A. Baird, L. Stappen *et al.*, "The interspeech 2021 computational paralinguistics challenge: Covid-19 cough, covid-19 speech, escalation & primates," *arXiv preprint arXiv:2102.13468*, 2021.

[13] M. Chen, X. He, J. Yang, and H. Zhang, "3-d convolutional recurrent neural networks with attention model for speech emotion recognition," *IEEE Signal Processing Letters*, vol. 25, no. 10, pp. 1440–1444, 2018.

[14] B. Schuller, S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani, F. Weninger, F. Eyben, E. Marchi *et al.*, "The interspeech 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism," in *Proceedings INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France*, 2013.

[15] B. Schuller, S. Steidl, A. Batliner, J. Epps, F. Eyben, F. Ringeval, E. Marchi, and Y. Zhang, "The interspeech 2014 computational paralinguistics challenge: Cognitive & physical load," 2014.

[16] C.-N. Anagnostopoulos, T. Iliou, and I. Giannoukos, "Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011," *Artificial Intelligence Review*, vol. 43, no. 2, pp. 155–177, 2015.

[17] P. Yenigalla, A. Kumar, S. Tripathi, C. Singh, S. Kar, and J. Vepa, "Speech emotion recognition using spectrogram & phoneme embedding." in *Interspeech*, 2018, pp. 3688–3692.

[18] A. Satt, S. Rozenberg, and R. Hoory, "Efficient emotion recognition from speech using deep learning on spectrograms." in *Interspeech*, 2017, pp. 1089–1093.

[19] F. Eyben, M. Wöllmer, B. Schuller, F. Weninger, M. Wollmer, and B. Schuller, "Opensmile: open-source media interpretation by large feature-space extraction," in *MM'10-Proceedings of the ACM Multimedia 2010 International Conference*, 2015.

[20] M. El Ayadi, M. S. Kamel, and F. Karray, "Survey on speech emotion recognition: Features, classification schemes, and databases," *Pattern recognition*, vol. 44, no. 3, pp. 572–587, 2011.

[21] S. Jing, X. Mao, and L. Chen, "Prominence features: Effective emotional features for speech emotion recognition," *Digital Signal Processing*, vol. 72, pp. 216–231, 2018.

[22] Y. Kim, H. Lee, and E. M. Provost, "Deep learning for robust feature generation in audiovisual emotion recognition," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 3687–3691.

[23] M. Sarma, P. Ghahremani, D. Povey, N. K. Goel, K. K. Sarma, and N. Dehak, "Emotion identification from raw speech signals using dnns." in *Interspeech*, 2018, pp. 3097–3101.

[24] S. Mirsamadi, E. Barsoum, and C. Zhang, "Automatic speech emotion recognition using recurrent neural networks with local attention," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 2227–2231.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[26] Y. Sakashita and M. Aono, "Acoustic scene classification by ensemble of spectrograms based on adaptive temporal divisions," *Detection and Classification of Acoustic Scenes and Events (DCASE) Challenge*, 2018.

[27] Y. Zeng, H. Mao, D. Peng, and Z. Yi, "Spectrogram based multi-task audio classification," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3705–3722, 2019.

[28] Z. Ren, J. Han, N. Cummins, and B. W. Schuller, "Enhancing transferability of black-box adversarial attacks via lifelong learning for speech emotion recognition models," *Proc. Interspeech 2020*, pp. 496–500, 2020.

[29] B.-H. Su, C.-M. Chang, Y.-S. Lin, and C.-C. Lee, "Improving speech emotion recognition using graph attentive bi-directional gated recurrent unit network," *Proc. Interspeech 2020*, pp. 506–510, 2020.

[30] Y. Xie, R. Liang, Z. Liang, C. Huang, C. Zou, and B. Schuller, "Speech emotion classification using attention-based lstm," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 11, pp. 1675–1685, 2019.

[31] M. Cornia, L. Baraldi, G. Serra, and R. Cucchiara, "Predicting human eye fixations via an lstm-based saliency attentive model," *IEEE Transactions on Image Processing*, vol. 27, no. 10, pp. 5142–5154, 2018.

[32] W. Chan and I. Lane, "Deep convolutional neural networks for acoustic modeling in low resource languages," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 2056–2060.

[33] T. N. Sainath, V. Peddinti, B. Kingsbury, P. Fousek, B. Ramabhadran, and D. Nahamoo, "Deep scattering spectra with deep neural networks for lvcsr tasks," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[34] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3474–3484.

[35] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.