INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Real-Time Causal Spectro-Temporal Voice Activity Detection Based on Convolutional Encoding and Residual Decoding

Jing-yuan Wang, Jie Zhang, Li-rong Dai

NERC-SLIP, University of Science and Technology of China, Hefei, China

jywg@mail.ustc.edu.cn, {jzhang6, lrdai}@ustc.edu.cn

Abstract

Voice activity detection (VAD) is an essential front-end in many speech applications that aims at determining the presence or absence of speech signals in an audio frame. However, traditional VAD methods often suffer from poor performance or non-causality in low signal-to-noise ratio (SNR) environments. In this work, we therefore present a real-time causal VAD model, which mainly consists of a frequency-domain feature generation module, a convolutional-based encoding module and a residual block based decoding module. The exploitation of only current and past frames for feature extraction guarantees the causality. The effectiveness of the proposed model is verified on two datasets under various noise conditions. It is shown that the proposed method can achieve a comparable or even better performance than state-of-the-art non-causal models.

Index Terms: Voice activity detection, residual network, convolutional network, causality.

1. Introduction

Voice activity detection (VAD) aims to identify the presence or absence of speech activities of interest in an audio signal that might be contaminated by various background noises. It is usually employed as a front-end preprocessor and largely affects the performance of back-end applications. For example, in automatic speech recognition (ASR), it was shown that even if the background noise is small, half of the word error rates are related to the front-end VAD mismatch [1]. In speech coding task, VAD can be exploited effectively for the reduction of the average bitrate and co-channel interference [2]. The application of VAD is also required by e.g., speech separation/enhancement, speaker diarization, etc.

Traditional VAD approaches predominantly rely on energy-based features, such as time-domain power [3], spectral feature [4], short-term energy [5] and spectral entropy [6]. However, in the case of low signal-to-noise ratios (SNRs), it becomes challenging to use traditional methods to distinguish human voice from noises. In order to improve the efficacy in low SNR conditions, some methods were thus proposed, e.g., time-frequency enhancement [7], denoising-based robust VAD (rVAD) [8]. More recently, with the advance in neural networks as well as the application to speech processing, fully connected deep networks [9, 10], convolutional neural networks (CNNs) [11–13], long short-term memory (LSTM) [14–16] and hybrid models [17–19] have been studied in this field. Although compared to conventional method the performance can be improved, they are still largely affected by the noise condition.

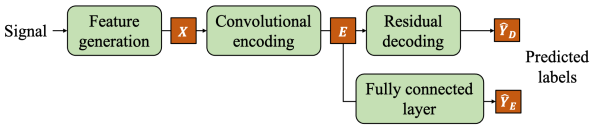

Figure 1: *The overall diagram of the proposed VAD model.*

To further address this limitation, some more complex models were proposed. For instance, Zhang and Wang [20] proposed to use multi-resolution cochleagram (MRCG) feature for training a bottleneck deep neural network (bDNN), which can achieve a better performance but require a much heavier computational burden. In [21], an attention-based adaptive context attention model (ACAM) was proposed, which utilizes contextual information during training and outperforms the bDNN. However, the training of ACAM is unstable. Lee et al. [22] proposed a spectro-temporal attention-based model (STAM) for VAD, which was built upon ACAM with the same feature inputs and can improve the training stability.

As the front-end VAD module is usually combined with much more complicated subsequent speech tasks, apart from the noise robustness an expected VAD model needs to further satisfy two requirements: 1) light-weighted in model size and 2) low latency. The model size is not only related to the space complexity, but also to the decoding time, which in addition determines the latency. For instance, in ACAM and STAM methods the use of contextual information makes the VAD non-causal, which has to wait for a few time frames to construct the input features. This non-causal design with a high latency is clearly not compatible with online speech tasks, e.g., streaming ASR.

In this paper, based on STAM [22] we therefore propose a real-time causal spectro-temporal VAD approach based on convolutional encoding and residual decoding. First, we calculate the log-Mel spectrograms of the input signal and construct a series of causal acoustic inputs, where only the current and past frames are included. The convolutional encoding module then processes these features using convolutional layers to highlight the informative parts. The residual decoding module utilizes residual connections to capture temporal dependency across time frames. The effectiveness of the proposed model is verified on two noisy datasets under various noise conditions, which shows a wider applicability of the proposed causal method. Compared to the state-of-the-art non-causal STAM method, our model can perform better in most cases with a smaller parameter amount. The rest of the paper is organized as follows. In Section 2, we describe the proposed causal VAD model in detail. Section 3 presents the experimental setup and

This work was supported by the National Natural Science Foundation of China (62101523), Hefei Municipal Natural Science Foundation (2022012) and USTC Research Funds of the Double First-Class Initiative (YD2100002008). (Correspondence: jzhang6@ustc.edu.cn)

10.21437/Interspeech.2023-716

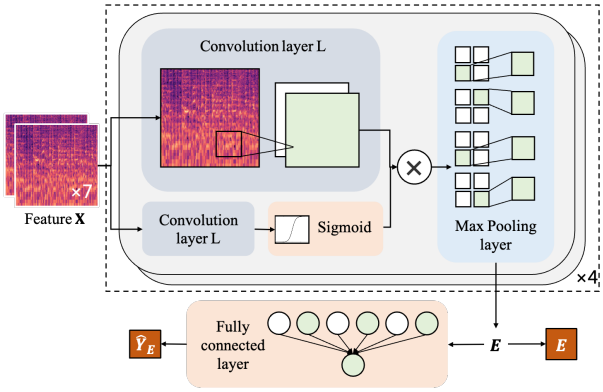

Figure 2: *An illustration of the convolutional encoding module.*

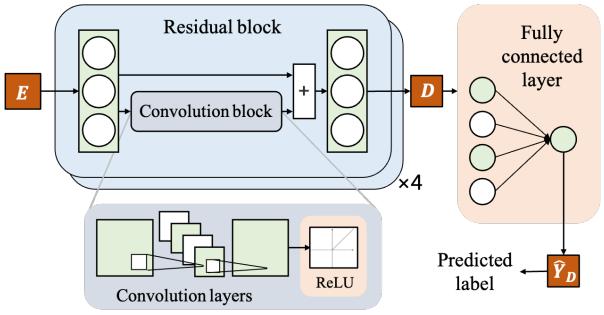

Figure 3: *An illustration of the residual decoding module.*

evaluation results. Finally, Section 4 concludes this work.

2. Proposed Method

The proposed VAD model mainly comprises three modules: feature generation, convolutional encoding and residual decoding, which follow the pipeline in Figure 1.

2.1. Feature generation module

In order to generate the necessary features and labels for the proposed model, the original speech signal is initially segmented using a Hann window of 25 ms with a shift of 10 ms and then converted using a 1024-point short-time Fourier transform (STFT). The STFT-domain frames are then filtered using a log-Mel filter bank [23] with a factor of $D = 80$, resulting in the input for the model. Similarly to [21], the features and labels for the model are constructed from a series of frames, that is, the feature vector at time index T incorporates the information from both the current and past frames, given by

$$\boldsymbol{X} = [\boldsymbol{F}_{T-t_0}, \boldsymbol{F}_{T-t_1}, \boldsymbol{F}_{T-t_2}, \cdots, \boldsymbol{F}_{T-t_n}]^\top, \quad (1)$$

where $\boldsymbol{t} = [t_0, t_1, t_2, \cdots, t_n]$ denotes the set of relative time indices of the considered frames, and $\boldsymbol{F}$ represents the frame-level log-Mel spectrogram features. Similarly, the ground-truth label vector at the current time step is defined as

$$\boldsymbol{L} = [L_{T-t_0}, L_{T-t_1}, L_{T-t_2}, \cdots, L_{T-t_n}]^\top, \quad (2)$$

where $(\cdot)^\top$ represents the vector/matrix transpose. It is clear from the considered feature formation that when detecting the status of the current frame, only the current and existing information is used and the future frames are not required, leading to the causality of the proposed VAD model.

2.2. Convolutional encoding module

In this work, the proposed convolutional layer in convolutional encoding module is based on the gated CNN-based spectral attention module in [22]. Although the batch normalization layer in DNNs can accelerate the model convergence and mitigate the gradient dispersion issue, the layer number of the convolutional encoding module is shallow, and it is validated via experiments that excluding the batch normalization layer does not heavily affect the performance. Therefore, we remove the batch normalization layer to reduce the parameter amount.

In detail, the convolutional encoding module is comprised of multiple convolution layers and a max-pooling layer, as shown by the dotted box in Figure 2. Given the input acoustic feature $\boldsymbol{X}$, the upper convolution layer is responsible for extracting relevant feature $\boldsymbol{C_X}$, and the lower convolution layer generates the corresponding mask weight $\boldsymbol{M_X}$. The acoustic features $\boldsymbol{X}$ and the mask matrix $\boldsymbol{C_X}$ are multiplied and then fed into the max-pooling layer. The output of the max-pooling layer $\boldsymbol{E}$ is fed into the fully connected layer to obtain the predicted output $\hat{\boldsymbol{Y}}_{\boldsymbol{E}}$ for loss calculation during training. It is clear that $\hat{\boldsymbol{Y}}_{\boldsymbol{E}}$ can be regarded as a raw VAD prediction. The kernels in the convolution layers have a size of 3×3. The input and output channels increase from $\{1, 2\}$ to $\{8, 16\}$ in order according to the number of layers. That is, in case the layer index (four in total) is increased by one, the input and output channels of the corresponding layer is increased by two. The pooling layer has a window of 2×2. The numbers of hidden unit and output unit in the fully connected layer are 256 and 1, respectively. The number of layers in convolutional encoding module is empirically set to be 4 in this work, which is shown to be effective in performance.

2.3. Residual decoding module

We design a residual decoding module to decode the encoded feature, which comprises a residual convolution block [24] and a fully connected layer as shown in Figure 3. In the convolutional encoding module, the prediction label $\hat{\boldsymbol{Y}}_{\boldsymbol{E}}$ is the output of only one full-connection layer given the input $\boldsymbol{E}$. This means that $\boldsymbol{E}$ already has some semantic information, and it is expected to retain this information as much as possible in the decoder. To accomplish this, residual connections are used throughout the decoding module, which can help to retain information from the previous stage and only train the error module of the feature information with the actual label. With multi-layer stacking, the convolution blocks allow for a large receptive field with a small number of parameters due to the nature of the convolution kernel. Note that we configure four residual convolution blocks in the feature decoding module.

The output of the max-pooling layer $\boldsymbol{E}$ is fed into the first residual block, whose output will be forwarded to the next residual block. This operation is repeated in every residual block until the output $\boldsymbol{D}$ is obtained. Each residual block essentially has two convolution layers. The output of each residual block is the sum of the output of the convolution block and the input of this residual block. Finally, the output of residual blocks $\boldsymbol{D}$ is fed into the fully connected layer to obtain the predicted output $\hat{\boldsymbol{Y}}_{\boldsymbol{D}}$. The convolution kernels in the convolution layers have a size of 3×3, and the numbers of channels in each convolution layers are $\{1, 4, 1\}$, while the number of output unit in the fully connected layer is 1. The number of residual blocks is also set to be 4 similarly to the convolutional encoding module.

2.4. Loss function

In this work, we use both the output of the convolutional encoding module $\hat{\boldsymbol{Y}}_{\boldsymbol{E}}$ and the output of the residual decoding module $\hat{\boldsymbol{Y}}_{\boldsymbol{D}}$ to calculate the binary cross entropy (CE) loss with respect to the ground truth, which is given by

$$\mathcal{L} = (1-k)CE[\boldsymbol{L}, S(\hat{\boldsymbol{Y}}_{\boldsymbol{E}})] + kCE[\boldsymbol{L}, S(\hat{\boldsymbol{Y}}_{\boldsymbol{D}})], \quad (3)$$

where CE means the binary CE loss function and S the sigmoid activation function, and the hyper-parameter k is used to assign the importances of the convolutional encoding and residual decoding modules. Notably, all frames are taken into account in the loss calculation. In experiments, k is set to be 0.7 unless stated elsewhere.

3. Experiments

In this section, we will present ablation experiments to show the efficacy of each module in the proposed model as well as the comparison to some other existing models. We conducted experiments on the Intel Xeon E5-2680 CPU and NVIDIA GeForce GTX 3090 GPU. All experiments are conducted using the Adam optimizer [25] with the learning rate changing from 10^{-3} to 10^{-5}, and the decay proportion of the learning rate at each epoch is 0.8 with respect to the previous step.

3.1. Experimental setup

Dataset: Two datasets, including QUT-NOISE-TIMIT [26] and LibriSpeech [27], are utilized to validate the VAD performance in this work. The QUT-NOISE-TIMIT dataset was formed by combining the TIMIT dataset [28] with the QUT-NOISE background noise dataset [26]. This mixing operation leads to noisy speech data with a cumulative duration of 600 hours across ten scenes at six different SNR levels of (-10, -5, 0, 5, 10, 15) dB. To maintain the independence of the training and testing sets, 100 hours of data were randomly selected as the training subset and another 100 hours as the test subset. Note that the training and test sets have no coincident segments in terms of either human voice or background noise.

As LibriSpeech was mainly collected for the English ASR task, we mix the LibriSpeech dev-clean dataset [27] with NoiseX-92 [29] dataset to create another noisy test dataset, which contains 15 hours of audio recordings across 15 environments and 6 different SNR levels of (-10, -5, 0, 5, 10, 15) dB. Since our focus is on the VAD in low SNR conditions, the results of SNR $\in \{$-10, -5, 0, 5$\}$ dB will be shown.

Performance measure: We use the area under the curve (AUC) [30] to measure the VAD accuracy, which is frequently employed in classification problems and denotes the area under the curve of the receiver operating characteristic (ROC) [31]. In addition, we use the THOP package to calculate the parameter quantity and the calculation quantity in flops, which measures the space and computational complexities, respectively.

3.2. Experimental results

First of all, as the frame combination determines the contextual feature extraction, we compare the performance of using continuous and non-continuous frames. Notice that given a fixed number of frames, the exploitation of continuous frames results in a shorter time span, which may limit the amount of temporal information that can be captured; the non-continuous frames can increase the time span but introduce the temporal discontinuity. For this, we consider $\boldsymbol{t} = [t_0, t_1, t_2, ..., t_n]$ = [0, 1, 2, 3, 4, 5, 6] as the continuous frame set and $\boldsymbol{t}$ = [0, 1, 3, 7, 15, 25, 38] as the non-continuous set, which follows the frame span in [13] (i.e., t_0 = 0, t_n = 38). From Table 1, we can see that on the QUT-NOISE-TIMIT dataset, the utilization of non-continuous frame set outperforms the continuous counterpart. This implies that the time span plays a more important role than time continuity in VAD, as a longer time span captures more contextual information. Hence, we will only consider the non-continuous combination of time frames in the sequel.

Table 1: *The AUC of ablation experiments.*

SNR (dB)		-10	-5	0
-ResDec	continuous	83.48	90.70	95.66
	non-continuous	88.13	95.48	98.80
ResFC	continuous	83.45	90.62	95.54
	non-continuous	89.08	96.15	99.00
ResConv	continuous	83.45	90.41	95.36
	non-continuous	**90.31**	**96.64**	**99.09**

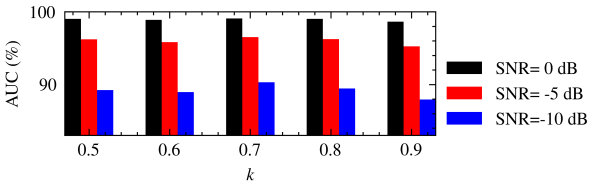

Figure 4: *The AUC in terms of k and SNR.*

Second, as using the output of the convolutional encoding module can somehow directly predict the raw labels through the fully connected layer, it is necessary to compare the effect of different modules on the performance. To do so, we compare three cases: the network without residual decoding (-ResDec), using the residual fully connected layers (ResFC) and using the proposed residual convolutional layers (ResConv). The obtained results on the QUT-NOISE-TIMIT dataset are presented in Table 1, from which it is clear that using the proposed residual convolutional layers in the residual decoding module achieves the best performance. This is because the receptive field of the CNN is already capable of covering all input frames, and the convolutional kernel is capable of efficiently learning the information contained in the features.

Further, we investigate the impact of the hyper-parameter k in the loss function on the VAD performance. In Figure 4, we show the AUC distribution in terms of k and SNR on the QUT-NOISE-TIMIT dataset. It is clear that in case k = 0.7, the best detection performance is achieved regardless of the noise condition (this becomes more clear in case of SNR = -10 dB). A larger or smaller k will worsen the AUC. Also, it conveys that k plays a more important role in lower SNR conditions.

Third, we compare the performance of the proposed method on the QUT-NOISE-TIMIT and noisy LibriSpeech datasets with existing state-of-the-art approaches, including ACAM [21], STAM [22]. Note that ACAM and STAM use non-causal inputs, which require a look ahead of 19 frames. To ensure the consistency of data input, we use log-Mel spectrogram features of seven frames for all models, where the causal method (i.e., the proposed method) use the relative frame set $\boldsymbol{t}$ =

https://github.com/Lyken17/pytorch-OpCounter

Table 2: *The AUC score on the QUT-NOISE-TIMIT dataset.*

SNR (dB)	-10	-5	0	5
ACAM [21]	86.22	94.32	98.32	99.20
STAM [22]	88.40	95.78	98.84	99.53
Ours	**90.31**	**96.64**	**99.09**	**99.64**

Table 3: *The AUC score on the noisy LibriSpeech dataset.*

SNR (dB)	-10	-5	0	5
ACAM [21]	92.82	96.30	97.74	97.63
STAM [22]	93.56	96.03	97.27	97.76
Ours	**94.48**	**96.82**	**97.80**	**98.16**

Table 4: *The model complexity of comparison VAD methods.*

Model	FLOPs	parameters	Causality
ACAM [21]	32.2M	328K	No
STAM [22]	41.9M	559K	No
Ours	39.8M	**360K**	Yes

[0, 1, 3, 7, 15, 25, 38] and the non-causal methods (STAM and ACAM) use $\boldsymbol{t}$ = [-19, -10, 1, 0, 1, 10, 19] (this keeps the same as in [21, 22]) with respect to the current frame. Tables 2&3 present the comparisons on the two datasets, respectively. It is clear on both datasets that the proposed method achieves the best performance regardless of noise levels, which is slightly better than that of existing best non-causal STAM model. This is mainly due to the fact that the proposed convolutional encoding module enables a raw VAD prediction and the residual decoding provides a further label refinement. From Table 4, we observe that compared to the best published non-causal STAM model, apart from the superiority in performance the proposed method even introduces less FLOPs and parameters, showing a wider applicability.

Finally, a more complete comparison with the state-of-the-art STAM model [22] on the QUT-NOISE-TIMIT dataset is shown in Table 5, where several types of additive noises and noise levels in SNR are taken into account. It can be seen that the proposed method works much better than the non-causal STAM in *CAFE-FOODCOURTB, REVERB-CARPARK, REVERB-POOL*, where note that the latter two denote reverberant environments, and in the rest conditions the two methods perform comparably. Notice that in the two reverberant environments, the AUC of the proposed model is 2.8% higher than that of STAM on average, showing a stronger robustness against reverberations. Note that the real-time factor of the proposed method on the QUT-NOISE-TIMIT dataset is approximately 0.03, which is acceptable for real-time applications. Overall, compared to STAM, the proposed method can not only resolve the issue of non-causality, but also improve the VAD accuracy by 0.74% on average.

4. Conclusion

In this paper, we proposed a real-time causal spectro-temporal VAD model based on convolutional encoding and residual decoding. The input acoustic features were constructed using log-mel spectrograms of the current and previous time frames, forming a causality property. Experiments on two noisy datasets validated the efficacy of the proposed method. Compared to existing causal methods, our method can achieve a much higher AUC; compared to the more advanced non-causal approaches our method obtains a comparable or even better performance with a lower complexity, particularly in low SNR and reverberant conditions. Therefore, the proposed method is more appropriate for real-time applications, e.g., streaming ASR.

Table 5: *The comparison of AUC with the best existing STAM method on the QUT-NOISE-TMIMT dataset with different noise levels and types.*

Noise type	SNR	STAM [22]	Ours
CAFE-CAFE	-10	78.38	**79.07**
	-5	91.85	**92.05**
	0	97.83	97.67
CAFE-FOODCOURTB	-10	73.76	**76.56**
	-5	87.29	**90.56**
	0	96.42	**97.23**
CAR-WINDOWNB	-10	97.37	97.21
	-5	99.21	98.91
	0	99.77	99.45
CAR-WINUPB	-10	99.16	98.85
	-5	99.73	99.38
	0	99.87	99.59
HOME-KITCHEN	-10	97.59	97.41
	-5	99.10	98.75
	0	99.64	99.28
HOME-LIVINGB	-10	92.06	**92.59**
	-5	96.96	**97.12**
	0	99.10	98.94
REVERB-CARPARK	-10	89.46	**91.04**
	-5	95.53	**96.22**
	0	98.64	**98.71**
REVERB-POOL	-10	74.28	**82.34**
	-5	88.12	**91.97**
	0	95.80	**96.78**
STREET-CITY	-10	96.76	**97.16**
	-5	99.09	98.77
	0	99.57	99.40
STREET-KG	-10	94.31	**95.95**
	-5	98.22	**98.26**
	0	99.57	99.26
Average	-5	94.48	**95.22**

5. References

[1] M. H. Savoji, "A robust algorithm for accurate endpointing of speech signals," *Speech communication*, vol. 8, no. 1, pp. 45–60, 1989.

[2] A. Gersho and E. Paksoy, "An overview of variable rate speech coding for cellular networks," in *IEEE Int. Conf. on Selected Topics in Wireless Communications*, 1992, pp. 172–175.

[3] L. R. Rabiner and M. R. Sambur, "An algorithm for determining the endpoints of isolated utterances," *Bell System Technical Journal*, vol. 54, no. 2, pp. 297–315, 1975.

[4] J. Haigh and J. Mason, "Robust voice activity detection using cep-

stral features," in *Proc. of Int. Conf. on Computers, Communciations and Automation*, vol. 3, 1993, pp. 321–324.

[5] R. Zhang and H. Cui, "Speech endpoint detection algorithm analyses based on short-term energy," *Audio Engineering*, vol. 7, p. 015, 2005.

[6] L. Jin and J. Cheng, "An improved speech endpoint detection based on spectral subtraction and adaptive sub-band spectral entropy," in *Int. Conf. on Intelligient Computation Techonolgy and Automation*, vol. 1, 2010, pp. 591–594.

[7] J. Sohn, N. S. Kim, and W. Sung, "A statistical model-based voice activity detection," *IEEE Signal Process. Letters*, vol. 6, no. 1, pp. 1–3, 1999.

[8] Z. Tan, K. Achintya, and N. Dehak, "rVAD: An unsupervised segment-based robust voice activity detection method," *Computer speech & language*, vol. 59, pp. 1–21, 2020.

[9] Y. Jung, Y. Kim, H. Lim, and H. Kim, "Linear-scale filterbank for deep neural network-based voice activity detection," in *Conf. of the Oriental Chapter of the Int. Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*, 2017, pp. 1–5.

[10] Y. Jung, Y. Kim, Y. Choi, and H. Kim, "Joint learning using denoising variational autoencoders for voice activity detection." in *Interspeech*, 2018, pp. 1210–1214.

[11] D. A. Silva, J. A. Stuchi, R. P. V. Violato, and L. G. D. Cuozzo, "Exploring convolutional neural networks for voice activity detection," *Cognitive technologies*, pp. 37–47, 2017.

[12] A. Sehgal and N. Kehtarnavaz, "A convolutional neural network smartphone app for real-time voice activity detection," *IEEE Access*, vol. 6, pp. 9017–9026, 2018.

[13] F. Jia, S. Majumdar, and B. Ginsburg, "Marblenet: Deep 1d time-channel separable convolutional neural network for voice activity detection," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2021, pp. 6818–6822.

[14] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with LSTM recurrent neural networks and an application to hollywood movies," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2013, pp. 483–487.

[15] J. Kim, J. Kim, S. Lee, J. Park, and M. Hahn, "Vowel based voice activity detection with LSTM recurrent neural network," in *Proceedings of the 8th International Conference on Signal Processing Systems*, 2016, pp. 134–137.

[16] P. Sertsi, S. Boonkla, V. Chunwijitra, N. Kurpukdee, and C. Wutiwiwatchai, "Robust voice activity detection based on LSTM recurrent neural networks and modulation spectrum," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2017, pp. 342–346.

[17] Z. Xueying, N. Puhua, and G. Fan, "DNN-LSTM based VAD algorithm," *Journal of Tsinghua University (Science and Technology)*, vol. 58, no. 5, pp. 509–515, 2018.

[18] A. Vafeiadis, E. Fanioudakis, I. Potamitis, K. Votis, D. Giakoumis, D. Tzovaras, L. Chen, and R. Hamzaoui, "Two-dimensional convolutional recurrent neural networks for speech activity detection," in *Interspeech*, 2019, pp. 2045—-2049.

[19] N. Wilkinson and T. Niesler, "A hybrid CNN-BiLSTM voice activity detector," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2021, pp. 6803–6807.

[20] X. Zhang and D. Wang, "Boosting contextual information for deep neural network based voice activity detection," *IEEE/ACM Trans. on Audio, Speech, and Language Process.*, vol. 24, no. 2, pp. 252–264, 2015.

[21] J. Kim and M. Hahn, "Voice activity detection using an adaptive context attention model," *IEEE Signal Process. Letters*, vol. 25, no. 8, pp. 1181–1185, 2018.

[22] Y. Lee, J. Min, D. K. Han, and H. Ko, "Spectro-temporal attention-based voice activity detection," *IEEE Signal Process. Letters*, vol. 27, pp. 131–135, 2019.

[23] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Audio, Speech, Language Process.*, vol. 28, no. 4, pp. 357–366, 1980.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of IEEE Conf. on Computer Vision Pattern Recognition (CVPR)*, 2016, pp. 770–778.

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

[26] D. Dean, S. Sridharan, R. Vogt, and M. Mason, "The QUT-NOISE-TIMIT corpus for evaluation of voice activity detection algorithms," in *Interspeech*, 2010, pp. 3110–3113.

[27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2015, pp. 5206–5210.

[28] V. W. Zue and S. Seneff, "Transcription and alignment of the TIMIT database," in *Recent Research Towards Advanced Man-Machine Interface Through Spoken Language*, 1996, pp. 515–525.

[29] A. Varga and H. J. Steeneken, "Assessment for automatic speech recognition: Ii. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech communication*, vol. 12, no. 3, pp. 247–251, 1993.

[30] J. Myerson, L. Green, and M. Warusawitharana, "Area under the curve as a measure of discounting," *Journal of the experimental analysis of behavior*, vol. 76, no. 2, pp. 235–243, 2001.

[31] M. H. Zweig and G. Campbell, "Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine," *Clinical chemistry*, vol. 39, no. 4, pp. 561–577, 1993.