Interspeech 2024
1-5 September 2024, Kos, Greece

Speaker Conditional Sinc-Extractor for Personal VAD

En-Lun Yu[1], *Kuan-Hsun Ho*[1], *Jeih-weih Hung*[2], *Shih-Chieh Huang*[3], *Berlin Chen*[1]

[1]National Taiwan Normal University, Taiwan
[2]National Chi Nan University, Taiwan
[3]Realtek Semiconductor Corp., Taiwan

[1]{enlunyu, jasonho610, berlin}@ntnu.edu.tw, [2]jwhung@ncnu.edu.tw,
[3]eric.sc.huang@realtek.com

Abstract

This study explores Sinc-convolution's novel application in Personal Voice Activity Detection (PVAD). The Sinc-Extractor (SE) network, developed for PVAD, learns cutoff frequencies and band gains of sinc functions to extract acoustic features. Additionally, the speaker conditional SE (SCSE) module incorporates speaker information from high-dimensional d-vectors into low-dimensional acoustic features. SE-PVAD and Vanilla PVAD have similar model size and computing load, while SCSE-PVAD is more compact with shorter inference time as it excludes speaker embedding. Evaluated with concatenated utterances from the LibriSpeech corpus, SE-PVAD outperforms Vanilla PVAD significantly. SCSE-PVAD matches Vanilla PVAD's performance but reduces input feature dimensionality and network complexity. Thus, SCSE-PVAD can function like a typical VAD, accepting only acoustic features, making it suitable for low-resource wearable devices.

Index Terms: voice activity detection, personalized voice activity detection, sinc-convolution

1. Introduction

With the recent proliferation of wearable devices, the usage scenarios for personalized voice functionalities have become increasingly complex. For instance, when wearing earbuds, Occlusion Effect Cancellation (OEC) [1, 2] is employed to reduce the user's own noise by utilizing signals within the ear canal. However, in devices like True Wireless Stereo (TWS) with a hear-through function [3, 4], feedback signals may complicate the signal components received by in-ear microphones, affecting OEC performance. To address this issue, detecting the user's voice activity becomes imperative. While Own Voice Detection [5, 6] serves this purpose, such methods typically require additional sensors on the device to enhance performance. However, this option is not always feasible as it leads to an increase in the production costs of the device. Therefore, within hardware limitations, Voice Activity Detection (VAD) tailored to the wearer emerges as another solution pathway.

VAD aims to categorize audio segments into speech or non-speech and plays a pivotal role in real-world applications in the context of numerous burgeoning speech-processing tasks such as speaker verification [7, 8], emotion estimation [9], and speech recognition [10]. Given the high-performance demands of these tasks, they often come with higher computational costs. VAD commonly serves as filtering modules for downstream tasks, improving the performance of downstream modules by ignoring non-speech signals and reducing overall computing costs.

Standard VAD [11, 12, 13] indiscriminately detects speech segments, regardless of the speaker, making it unsuitable for personalized contexts. Consequently, there is an urgent demand for VAD systems targeting specific speakers. Addressing this need, Shaojin Ding *et al.* proposed Personal VAD (PVAD) [14, 15], a system that identifies a particular speaker's speech activity at the frame-level. PVAD concatenates acoustic features and speaker embeddings such as d-vectors [16, 17] or i-vectors [18] to recognize whether the input frame corresponds to the target speaker's speech. In implementation, users only require a simple enrollment beforehand. Numerous studies [19, 20, 21, 22] have showcased PVAD's feasibility. Its key advantage lies in its ability to filter out noise unrelated to the target speaker, thereby reducing false alarms and recognition errors in downstream tasks. Therefore, PVAD is more suitable than Standard VAD for personalized wearable devices.

Originally, PVAD employed hand-crafted acoustic features such as Mel-FBANK. Although such features align with human perception, they may not be optimal for speech-related tasks. Along this direction, SincNet [23] proposed a Sinc-convolution that constrains the shape of filters and learns task-relevant filter banks while maintaining higher interpretability. Previously, [24] introduced Sinc convolution as an Encoder into the time-domain standard VAD to obtain features from different frequency bands. However, it is challenging to intuitively combine time-domain acoustic features with frequency-domain speaker embedding features to develop PVAD. Therefore, how to combine Sinc convolution and PVAD that considers both time and frequency domains has become an intriguing question.

This study proposed a simple but novel feature extraction method for PVAD called Speaker Conditional Sinc-Extractor (SCSE). Inspired by [23], Sinc Extractor (SE) in SCSE employs learnable filter banks instead of conventional hand-crafted ones. Moreover, SCSE uses speaker embedding feature to learn the frequency variations among speakers by linearly transforming the band gain and cutoff frequency of filter banks. That enables the speaker's information to be directly reflected in the extracted features. The experimental results demonstrate that SE-fueled PVAD (SE-PVAD) significantly outperforms Vanilla PVAD in performance while maintaining almost the same model size. On the other hand, SCSE greatly reduces the model size by incorporating the speaker embedding modulation mechanism into SE. Additionally, SE-PVAD and SCSE-PVAD inherit excellent convergence characteristics of SincNet. Even when trained with a small amount of data, it achieves better performance than Vanilla PVAD, which is a decisive advantage for data-driven PVAD models.

10.21437/Interspeech.2024-365

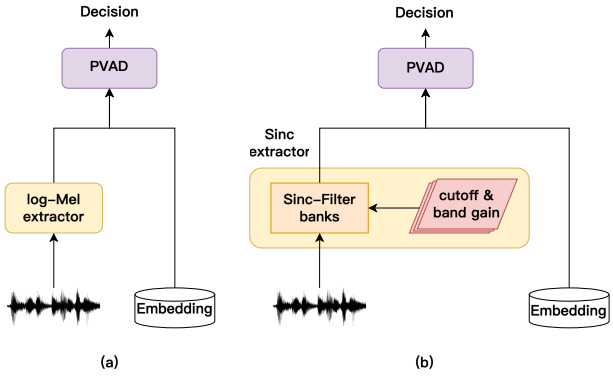

Figure 1: *PVAD with different feature extractor:(a) Vanilla PVAD using mel-filterbank. (b) SE-PVAD using sinc-filterbank.*

2. Methods

2.1. Vanilla PVAD

According to [14], Vanilla PVAD requires the target speaker to input their recordings into a pre-trained speaker recognition model, obtaining the target speaker embedding, denoted by $\mathbf{e}^{\text{target}}$ through encoding. This process is referred to as the enrollment process.

As depicted in Fig. 1(a), the Vanilla PVAD process begins by extracting the logarithmic Mel-filterbank energies $\{\tilde{\mathbf{x}}_t\}_{t=1}^{T}$ from the input signal $x[n]$ as the acoustic features, where $\tilde{\mathbf{x}}_t \in \mathbb{R}^{D\times 1}$ is the acoustic feature at time frame t, and T and D represent the total number of frames and the dimensionality of $\tilde{\mathbf{x}}_t$, respectively.

Subsequently, acoustic features of each frame t are concatenated with the target speaker embedding $\mathbf{e}^{\text{target}}$ obtained from the enrollment process:

$$\hat{\mathbf{x}}_t = [\tilde{\mathbf{x}}_t; \mathbf{e}^{\text{target}}]. \tag{1}$$

PVAD takes the concatenated features $\hat{\mathbf{x}}_t$ as input and predicts frame-level probabilities of three categories: target speaker's speech (tss), non-target speaker's speech (ntss), and non-speech (ns):

$$p_t = \text{PVAD}(\hat{\mathbf{x}}_t), \tag{2}$$

where $p_t = [p_t^{tss}, p_t^{ntss}, p_t^{ns}]$.

2.2. Sinc-based feature extractor

As described in Sec. 2.1, the acoustic feature $\tilde{\mathbf{x}}_t$ for Vanila PVAD is the logarithmic Mel-filterbank energies, which are obtained by applying the mel-filterbank to the frame-wise spectrum created by short-time Fourier transform (STFT), which detailed computation is as follows:

$$\tilde{\mathbf{x}}_{t,i} = \log(\sum_k (H_i[k]|X[k,t]|^2)), \quad i = 1, 2, ..., D, \tag{3}$$

where $\tilde{\mathbf{x}}_{t,i}$ is the i^{th} component of $\tilde{\mathbf{x}}_t$, $H_i[k]$ is the frequency response of i^{th} mel-filter, and $X[k,t]$ is the Fourier transform of $x_t[n]$, the input signal at time frame t. Even though mel-filter banks are designed to align with human perception, it has been demonstrated in [23] that they may not achieve optimal performance across all speech-related tasks. Therefore, based on the findings in [23], we propose a novel feature extraction method for PVAD called Sinc-Extractor (SE), as shown in Fig. 1(b). Specifically, SE employs sinc-filterbank as an alternative to mel-filterbank in Vanilla PVAD to extract sub-band energies as the acoustic features, and we term this new PVAD structure as SE-fueled PVAD (SE-PVAD).

Briefly speaking, SE produces acoustic features $\mathbf{x}_t \in \mathbb{R}^{D\times 1}$ at time frame t by computing the log-energies of sub-band signals, where each sub-band signal is the output of a sinc filter $h_i[n]$ with the input signal $x_t[n]$:

$$\mathbf{x}_{t,i} = \log(\sum_n (|x_t[n] * h_i[n]|^2)), \quad i = 1, 2, ..., D, \tag{4}$$

where $\mathbf{x}_{t,i}$ is the i^{th} component of $\mathbf{x}_t$, $x_t[n]$ is the input signal at time frame t, and "$*$" is the convolution operation. Referring to the reformed sinc-convolution framework in our previous work [25], the procedure to construct the sinc filterbank $\{h_i[n]\}_{t=1}^{D}$ is briefly described as follows:

First, the impulse response of the i^{th} sub-band sinc filter is initilaized as

$$\tilde{h}_i[n] = \frac{\omega_{c_2}^{(i)}}{\pi}\text{sinc}\left(\omega_{c_2}^{(i)} n\right) - \frac{\omega_{c_1}^{(i)}}{\pi}\text{sinc}\left(\omega_{c_1}^{(i)} n\right), \quad -\infty < n < \infty, \tag{5}$$

where $\omega_{c_1}^{(i)}$ and $\omega_{c_2}^{(i)}$ are low and high cutoff frequencies and the sinc function is defined as $\text{sinc}(x) = \sin(x)/x$. $\tilde{h}_i[n]$ is then delayed by M and truncated to be length $L = 2M + 1$, producing a causal FIR filter $\hat{h}_i[n] = \tilde{h}_i[n - M]$ for $0 \le n \le L - 1$ and $\hat{h}_i[n] = 0$ elsewhere. Finally, a learnable band gain factor b_i and the Hamming window function $w[n]$ is applied to $\hat{h}_i[n]$, producing its final version as

$$h_i[n] = b_i \cdot \hat{h}_i[n] \cdot w[n]. \tag{6}$$

In SE-PVAD, the sinc-filterbank feature $\mathbf{x}_t$ is employed to replace mel-filterbank feature $\tilde{\mathbf{x}}_t$ in Eq. (1) and concatenated with speaker embedding $\mathbf{e}^{\text{target}}$ to learn the PVAD model. Comparing Eqs. (3) and (4), mel-filterbank and sinc-filterbank work in frequency domain and time domain, respectively, to obtain sub-band energies. Since sinc-filterbank is learnable, its output is more likely to fit the specified task, such as PVAD here.

To sum up, the parameters to be learned for the SE module include $\{\omega_{c_1}^{(i)}, \omega_{c_2}^{(i)}, b_i\}_{i=1}^{D}$. In particular, we follow [25] to learn the cutoff frequencies with more freedom without violating the Nyquist theorem. That is, the cutoff frequencies are always lower than the half of the sampling rate.

2.3. Speaker Conditional Sinc-Extractor (SCSE)

SE is applied to highlight acoustic cues from the input signal for the PVAD task, but the target speaker information depends on the concatenated embedding $\mathbf{e}^{\text{target}}$ as shown in Eq. (1). The PVAD module has a large overall input feature size because of the direct concatenation of the relatively high-dimensional $\mathbf{e}^{\text{target}}$. Moreover, the target speaker embedding throughout the learning phase does not directly benefit the SE. With the intention of adapting sinc-filterbank in SE to target speakers, we suggest the Speaker Conditional Sinc-Extractor (SCSE) as a solution to these drawbacks. SCSE allows the PVAD to identify the target-speaker speech segments without directly utilizing the embedding $\mathbf{e}^{\text{target}}$, thus decreasing the input dimensionality. Furthermore, it prompts the SE module to determine the

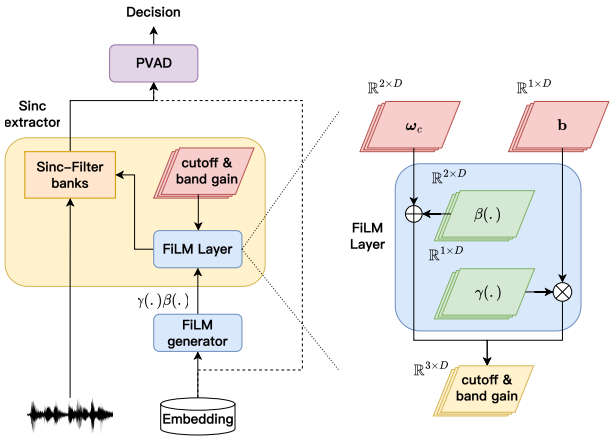

Figure 2: *Speaker Conditional Sinc-Extractor PVAD.* $\gamma(.)$ *and* $\beta(.)$ *are the the scaling and shifting vector of FiLM respectively.*

sub-bands of the input signal that are particularly relevant to the target speaker.

Fig. 2 depicts the architecture of SCSE-PVAD. SCSE performs an affine transformation on the cutoff frequencies and band gains of SE through a feature-wise linear modulation (FiLM) Layer [26], including biasing and scaling. Let $\boldsymbol{\omega}_c$ and $\mathbf{b}$ be the vectors that contain all cutoff frequencies and band gains, respectively. The speaker embedding $\mathbf{e}^{\text{target}}$ is employed as a representation of conditioning information, serving as the input of the FiLM generator to output a scaling vector $\boldsymbol{\gamma}(\mathbf{e}^{\text{target}})$ for $\boldsymbol{\omega}_c$ and a biasing vector $\boldsymbol{\beta}(\mathbf{e}^{\text{target}})$ for $\mathbf{b}$, both of which have dimensions matching $\boldsymbol{\omega}_c$ and $\mathbf{b}$ respectively. Subsequently, the parameters $\boldsymbol{\omega}_c$ and $\mathbf{b}$ of the Sinc-Extractor are tuned through the FiLM Layer:

$$[\tilde{\boldsymbol{\omega}}_c, \tilde{\mathbf{b}}] = \text{FiLM}(\boldsymbol{\omega}_c, \mathbf{b}) = [\boldsymbol{\omega}_c + \boldsymbol{\beta}(\mathbf{e}^{\text{target}}), \mathbf{b} \cdot \boldsymbol{\gamma}(\mathbf{e}^{\text{target}})]. \tag{7}$$

where $\tilde{\boldsymbol{\omega}}_c$ and $\tilde{\mathbf{b}}$ denote the updated cutoff frequencies and band gains of the speaker conditioned sinc-filterbank. Therefore, the PVAD module in SCSE-PVAD now only takes the acoustic features produced by SCSE as input without directly concatenating the embedding $\mathbf{e}^{\text{target}}$. Such an input feature dimension reduction can further simplify the network structure of PVAD.

2.4. Binary classification task in SCSE-PVAD

SCSE modifies the parameters of the sinc-filterbank through the FiLM layer conditioned with the embedding $\mathbf{e}^{\text{target}}$, which enables the direct reflection of target speaker information in the acoustic features. However, this means that the sinc-filterbank parameters vary according to the different target speakers during the inference stage, which could reduce the ability of the following PVAD module to identify non-target speaker's speech (ntss) from non-speech (ns). Thus, we convert the original PVAD's ternary classification task into a binary classification task to train SCSE-PVAD. This involves modifying PVAD to predict frame-level probabilities from $p_t = [p_t^{tss}, p_t^{ntss}, p_t^{ns}]$ to $p_t = [p_t^{tss}, p_t^{ntss\&ns}]$ in Eq. (2), where $p_t^{ntss\&ns} = 1 - p_t^{tss}$. This enables the SCSE module to concentrate on investigating sub-bands of input utterances that are dependent on the speaker, whereas the PVAD module operates more like a conventional VAD by merely predicting the probability of speech activity from the target speaker.

3. Experimental setup

3.1. Dataset

According to [14], no dataset might be available that covers the natural speaker turns and enrollment speech needed for the PVAD task. To conduct experiments, we followed [14] and used the LibriSpeech corpus [27] to prepare a dataset that contains concatenated utterances from multiple speakers. We randomly chose utterances from one to three speakers and concatenated them, with one randomly selected as the target speaker. To prevent model biasing, we included multiple-speaker utterances that did not contain the target-speaker speech with a probability of 0.2. While concatenated utterances cannot reproduce speech overlaps in real conversation scenarios, our experiments show that PVAD performs well in overlapping speech cases, consistent with the observations in [19].

The LibriSpeech training set consists of three subsets, totaling 960 hours of speech data from 2338 different speakers: train-clean-100 and train-clean-360 provide a total of 460 hours of clean speech, while train-other-500 offers 500 hours of noisy speech. Similarly, the test set of Librispeech contains clean and noisy utterances, totaling 10 hours from 73 speakers. We utilized the training and test sets from LibriSpeech respectively to create the training and test sets for the concatenated utterance dataset, comprising 140,000 and 5,500 utterances, respectively. Additionally, we employed the data augmentation technique from MTR [28], which introduces random noise sources with different Room Impulse Responses to prevent domain overfitting. Regarding speaker embedding, utterances from each speaker were randomly selected and fed into a pre-trained speaker verification model to generate window-level d-vectors. These d-vectors were then L2-normalized and averaged to produce the utterance-level d-vector, which serves as the target speaker embedding $\mathbf{e}^{\text{target}}$.

In order to evaluate PVAD, we consider not only the full training set (140,000 utterances) but also its subset, which is comprised of 10% of the total utterances. The experimental results will showcase the rapid convergence and exceptional performance of the presented SCSE-PVAD model, even when it is learned with a small training set.

3.2. Implementation details

Extracted from frames with a width of 25ms and a step of 10ms, the acoustic features with a size of $D = 40$ were generated using mel-filterbank and sinc-filterbank according to Eqs. (3) and (4). The length of each sinc-filter L is set to 251. Vanilla PVAD and SE-PVAD use a 2-layer LSTM network with 64 cells and a fully-connected layer with 64 neurons for their PVAD modules, whereas SCSE-PVAD uses a 2-layer LSTM network with 20 cells and a fully-connected layer with 20 neurons. The FiLM generator in SCSE-PVAD consists of a linear layer with a hyperbolic tangent activation function.

We used the Adam optimizer with a learning rate of 1×10^{-3} in the first epoch, which then decreased to 1×10^{-5} in subsequent epochs during training. The categorical cross-entropy function served as the loss function.

3.3. Metrics

The evaluation metrics for variant PVAD methods include the average precision (AP) for each class individually and the mean average precision (mAP) across all classes.

Table 1: *Architecture comparison results with training subset. BC: Binary classification. We report the Average Precision (AP) for each class and the mean AP (mAP) across all the classes.*

	BC	tss	ns	ntss	mAP
Vanilla PVAD (baseline)		0.494	0.805	0.495	0.599
SE-PVAD		**0.799**	**0.845**	**0.817**	**0.817**
SCSE-PVAD		0.614	0.746	0.634	0.658
SE-PVAD	✓	**0.837**	**0.918**		**0.888**
SCSE-PVAD	✓	0.773	0.829		0.776

Table 2: *Architecture comparison results with full training set. Parm.: the number of network parameters.*

	BC	tss	ns	ntss	mAP	Parm. (k)
Vanilla PVAD (baseline)		0.856	0.862	0.880	0.864	130.3
SE-PVAD		**0.929**	**0.902**	**0.932**	**0.925**	130.5
SCSE-PVAD	✓	0.823	0.909		0.878	**37.2**

4. Results and Discussions

The experiments are divided into two parts: First, we examine the performance of the PVAD models that were learned using a small training subset (10% of the total utterances in the full training set). Second, when trained with the full dataset, these PVAD models are assessed using VAD scores and the convergence speed.

4.1. Small Training Subset

Tab. 1 presents the AP and mAP scores of the test sets with various PVAD architectures trained on the small training set. From this table, we have the following observations:

1. As for the PVAD dealing with the ternary classification task, SE-PVAD provides the optimal AP scores for all three classes (ts, ntss, and ns), indicating that the acoustic features created by the learnable sinc-filterbank benefit PVAD a lot compared with those from mel-filterbank in Vanilla PVAD. SCSE-PVAD also outperforms Vanilla PVAD in mAP and AP specific to target speaker's speech (tss), which means that the 40-dim acoustic features from the speaker-conditional sinc-filterbank behave better than the 296-dim features (40-dim mel-filterbank features plus 256-dim speaker embedding) in PVAD when the the amount of training data is limited.
2. Both SE-PVAD and SCSE-PVAD can achieve better mAP and target-speaker AP scores when the PVAD switches from ternary to binary classification. These results show that acoustic features produced from the learnable sinc-filterbank can further highlight the target speaker's speech, aligning with the fundamental goal of PVAD.

4.2. Full Training Set

Tab. 2 includes the scores achieved by various PVADs learned with the full training dataset and the number of parameters in each PVAD architecture. By examining this table and comparing it with Tab. 1, we have the following discussions:

1. All of the three PVADs benefit from the increase of training data by achieving superior performance. Vanilla PVAD, in particular, improves the most as the training data increases, revealing the limitations of mel-filterbank acoustic features in the PVAD task.
2. SC-PVAD behaves significantly better than Vanilla PVAD and SCSE-PVAD, with tss-AP and mAP scores of 0.929 and 0.925, respectively. However, unlike the case with the small training subset, SCSE-PVAD obtains a lower tss-AP score than Vanilla PVAD when trained with the full dataset. The possible underlying reasons are: 1) It is challenging to effectively transmit speaker information from the d-vector representation to the acoustic features through a simple FiLM layer, and 2) the 40-dim acoustic features may not effectively represent the amount of speaker information dwelled in the 256-dimensional speaker embedding.
3. Despite its marginally worse performance, SCSE-PVAD has just 37.2k parameters and is the smallest of the three architectures. This advantage makes SCSE-PVAD better suited for on-device tasks.

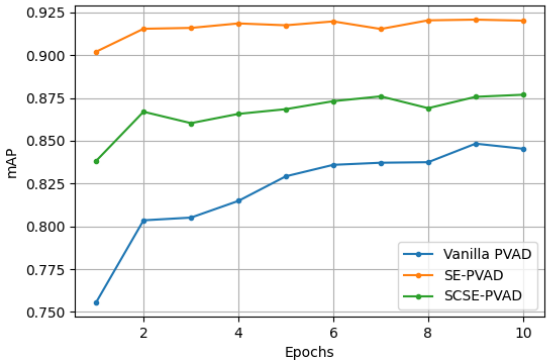

Figure 3: *Mean Average Precision (mAP) of different PVAD architecture with each epoch.*

Fig. 3 shows the mAP scores for the test set versus epochs during training for the three PVAD models. This figure demonstrates that sinc-filterbank is superior to mel-filterbank for preparing the acoustic features of the PVAD task, as both SE-PVAD and SCSE-PVAD converge in significantly fewer epochs than Vanilla PVAD.

5. Conclusions

In this study, we proposed a simple yet effective preprocessing strategy for PVAD called Sinc-Extractor (SE). SE learns and generates acoustic features that are aligned with the PVAD task objectives. As an extension, we also introduced the Speaker Conditional Sinc-Extractor (SCSE), which incorporates the speaker embedding modulation technique into SE. Experimental results demonstrate that SE significantly improves the performance of PVAD by building highly-effective acoustic features. On the other hand, although SCSE-PVAD outperforms Vanilla PVAD only in the case of small training subset, but its model size is approximately 30% that of Vanilla PVAD. In conclusion, we have demonstrated that using a learnable SE can achieve superior PVAD results with a negligible increase in model size. We have also found the feasibility of incorporating speaker conditions into SE and the advantages of its application to resource-limited devices.

6. Acknowledgement

This work was supported by Realtek Semiconductor Corporation. Any findings and implications in the paper do not necessarily reflect those of the sponsor.

7. References

[1] S. Liebich and P. Vary, "Occlusion Effect Cancellation in Headphones and Hearing Devices—The Sister of Active Noise Cancellation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, pp. 35–48, 2022.

[2] R. C. Borges and M. H. Costa, "A feed forward adaptive canceller to reduce the occlusion effect in hearing aids," *Computers in Biology and Medicine*, pp. 266–275, 2016.

[3] F. Denk, M. Hiipakka, B. Kollmeier, and S. M. A. Ernst, "An individualised acoustically transparent earpiece for hearing devices," *International Journal of Audiology*, pp. S62–S70, 2018.

[4] P. Hoffmann, F. Christensen, and D. Hammershøi, "Insert earphone calibration for hear-through options," *Proceedings of the AES International Conference*, pp. X105–112, 2013.

[5] P. Pertila, E. Fagerlund, A. Huttunen, and V. Myllyla, "Online Own Voice Detection for a Multi-Channel Multi-Sensor In-Ear Device," *IEEE Sensors J.*, pp. 27 686–27 697, 2021.

[6] F. Lindstrom, K. Ren, H. Li, and K. P. Waye, "Comparison of Two Methods of Voice Activity Detection in Field Studies," *J Speech Lang Hear Res*, pp. 1658–1663, 2009.

[7] T. Kinnunen and P. Rajan, "A practical, self-adaptive voice activity detector for speaker verification with noisy telephone and microphone data," *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 7229–7233, 2013.

[8] Y. Jung, Y. Choi, and H. Kim, "Self-Adaptive Soft Voice Activity Detection using Deep Neural Networks for Robust Speaker Verification," *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 365–372, 2019.

[9] F. Ringeval, S. Amiriparian, F. Eyben, K. Scherer, and B. Schuller, "Emotion Recognition in the Wild: Incorporating Voice and Lip Activity in Multimodal Decision-Level Fusion," *Proceedings of the 16th International Conference on Multimodal Interaction*, pp. 473–480, 2014.

[10] T. Yoshimura, T. Hayashi, K. Takeda, and S. Watanabe, "End-to-End Automatic Speech Recognition Integrated With CTC-Based Voice Activity Detection," 2020.

[11] 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," *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 483–487, 2013.

[12] S.-Y. Chang, B. Li, T. N. Sainath, G. Simko, and C. Parada, "Endpoint Detection Using Grid Long Short-Term Memory Networks for Streaming Speech Recognition," *Interspeech 2017*, pp. 3812–3816, 2017.

[13] M. Shannon, G. Simko, S.-Y. Chang, and C. Parada, "Improved End-of-Query Detection for Streaming Speech Recognition," *Interspeech 2017*, pp. 1909–1913, 2017.

[14] S. Ding, Q. Wang, S.-y. Chang, L. Wan, and I. L. Moreno, "Personal VAD: Speaker-Conditioned Voice Activity Detection," 2020.

[15] S. Ding, R. Rikhye, Q. Liang, Y. He, Q. Wang, A. Narayanan, T. O'Malley, and I. McGraw, "Personal VAD 2.0: Optimizing Personal Voice Activity Detection for On-Device Speech Recognition," 2022.

[16] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Trans. Audio Speech Lang. Process.*, pp. 788–798, 2011.

[17] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized End-to-End Loss for Speaker Verification," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4879–4883, 2018.

[18] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4052–4056, 2014.

[19] N. Makishima, M. Ihori, T. Tanaka, A. Takashima, S. Orihashi, and R. Masumura, "Enrollment-less training for personalized voice activity detection," *Interspeech 2021*, 2021.

[20] I. Medennikov, M. Korenevsky, T. Prisyach, Y. Khokhlov, M. Korenevskaya, I. Sorokin, T. Timofeeva, A. Mitrofanov, A. Andrusenko, I. Podluzhny, A. Laptev, and A. Romanenko, "Target-Speaker Voice Activity Detection: a Novel Approach for Multi-Speaker Diarization in a Dinner Party Scenario," *Proc. Interspeech*, pp. 274–278, 2020.

[21] A. Jayasimha and P. Paramasivam, "Personalizing Speech Start Point and End Point Detection in ASR Systems from Speaker Embeddings," *2021 IEEE Spoken Language Technology Workshop (SLT)*, pp. 771–777, 2021.

[22] M. He, D. Raj, Z. Huang, J. Du, Z. Chen, and S. Watanabe, "Target-speaker Voice Activity Detection with Improved I-Vector Estimation for Unknown Number of Speaker," *arXiv preprint arXiv:2108.03342*, 2021.

[23] M. Ravanelli and Y. Bengio, "Speaker Recognition from Raw Waveform with SincNet," *2018 IEEE Spoken Language Technology Workshop (SLT)*, pp. 1021–1028, 2018.

[24] O. Köpüklü and M. Taseska, "ResectNet: An Efficient Architecture for Voice Activity Detection on Mobile Devices," *Interspeech 2022*, pp. 5363–5367, 2022.

[25] K.-H. Ho, J.-w. Hung, and B. Chen, "What do neural networks listen to? Exploring the crucial bands in Speech Enhancement using Sinc-convolution," *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[26] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "FiLM: Visual Reasoning with a General Conditioning Layer," *AAAI*, 2018.

[27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.

[28] C. Kim, A. Misra, K. Chin, T. Hughes, A. Narayanan, T. N. Sainath, and M. Bacchiani, "Generation of Large-Scale Simulated Utterances in Virtual Rooms to Train Deep-Neural Networks for Far-Field Speech Recognition in Google Home," *Proc. Interspeech 2017*, pp. 379–383, 2017.