Learnable Spectro-temporal Receptive Fields for Robust Voice Type Discrimination

Tyler Vuong,[1] *Yangyang Xia,*[1] *Richard M. Stern*[1,2]

[1] Department of Electrical and Computer Engineering, Carnegie Mellon University, USA
[2] Language Technologies Institute, Carnegie Mellon University, USA

tvuong@andrew.cmu.edu, raymondxia@cmu.edu, rms@cs.cmu.edu

Abstract

Voice Type Discrimination (VTD) refers to discrimination between regions in a recording where speech was produced by speakers that are physically within proximity of the recording device ("Live Speech") from speech and other types of audio that were played back such as traffic noise and television broadcasts ("Distractor Audio"). In this work, we propose a deep-learning-based VTD system that features an initial layer of learnable spectro-temporal receptive fields (STRFs). Our approach is also shown to provide very strong performance on a similar spoofing detection task in the ASVspoof 2019 challenge. We evaluate our approach on a new standardized VTD database that was collected to support research in this area. In particular, we study the effect of using learnable STRFs compared to static STRFs or unconstrained kernels. We also show that our system consistently improves a competitive baseline system across a wide range of signal-to-noise ratios on spoofing detection in the presence of VTD distractor noise.

Index Terms: voice type discrimination, spectro-temporal receptive field, spoofing detection, convolutional neural network

1. Introduction

The goal of Voice Type Discrimination (VTD) is to locate regions of "live speech" in an audio recording containing both "live speech" and "distractor audio." Live speech refers to speech segments spoken by speakers physically in the proximity of the recording device. Distractor audio is any other audio source that is not live speech such as environmental noises, television, and radio. This becomes challenging since distractor audio can contain natural human speech that is broadcast through a television or radio and often overlaps with the live speech in time. In addition, no assumption is made about the signal-to-noise ratio (SNR) or the geometry of the room where the recording took place, making methods that use auxiliary hardware to detect specific live speech patterns [1, 2] inapplicable. VTD is important for practical applications such as wake-up word detection for voice assistants [3]. Robust detection of voice under noise and music is also vital for defending against adversarial attack using nonspeech audio signals [4].

Carnegie Mellon University (CMU) was a participant in an assessment of the capabilities of VTD discrimination systems conducted in June and July of 2019 by the Johns Hopkins Applied Physics Laboratory (JHU/APL) using audio data that were recorded and annotated by SRI International (SRI)[5] in four rooms of different size and shape. We describe the latest CMU system in this paper.

The upper panel of Figure 1 depicts Room 3 used for the SRI data collection. The lower panel of that figure is a color-coded timeline that describes the actual presence and absence of the live speech and the various distractors. We note that live speech is present only a very small fraction of the total time.

Related work. While this work is part of the first effort ever to develop systematic approaches to voice type discrimination, VTD itself has some resemblances to voice activity detection (VAD) [6] and spoofing detection. Specifically, VTD can be thought of as a two-stage problem: (1) discrimination between speech and nonspeech, and (2) the separation of live speech from machine speech. The latter is obviously the far more challenging problem. Recent advancement in automatic detection of spoofing attacks have focused on specific forms of countermeasures in synthetic, converted, and replayed speech [7]. Related methods for each type of attack is therefore highly specialized. For example, Replay Attack (RA) countermeasures typically rely on detecting distortions in the higher-frequency bands (*e.g.* [8]). In VTD, however, we focus on more common distractor speech in TV and radio broadcasts. Despite being recorded in nature, broadcast speech contains a richer variety of speaking styles. In addition, the acoustical conditions for VTD are often more adverse than for spoofing detection.

The concept of the spectro-temporal receptive field (STRF) has been applied successfully in VAD and representation learning of speech. The form of the STRFs is motivated by physiological structures that are believed to exist in the central auditory system that respond to a range of patterns of temporal modulation and spectral modulation [9]. Mesgarani *et al.* proposed a STRF-based voice activity detector that is highly robust in the presence of excessive noise and reverberation [10]. More recently, Gabor-based modulation filters were implemented as kernels in a convolutional neural network (CNN) to aid representation learning for robust speech recognition [11]. In general, kernelizing CNN layers as filters has also proven useful in speaker and phoneme recognition [12, 13]. This work motivated us to build our VTD system around learnable STRFs.

Organization of this paper. In the next section we describe *STRFNet*, a system developed by CMU based on deep learning principles that was designed specifically to optimize performance on the 2019 JHU/APL VTD assessment. We then describe and discuss the performance of the system on VTD data provided by SRI. To complement the VTD task, we also evaluate our system on spoofing detection using the Logical Access (LA) data in the recent ASVspoof 2019 challenge [7].

2. The STRFNet System

Our design of the VTD system is motivated by three key assumptions: (1) the STRFs can reliably extract speech-specific features in the modulation domain under adverse conditions, (2) generic convolutional kernels are needed to extract patterns from the response to the STRF, and (3) long temporal patterns might reveal suprasegmental features such as prosody that

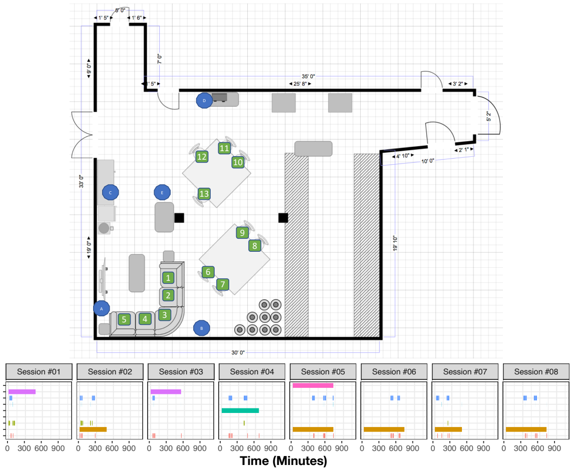

Figure 1: *Upper panel: Schematic diagram of Room 3, which was one of two rooms used for evaluation. The electret microphones are located at the blue circles, while the green squares indicate possible talker locations. Lower panel: Timeline of live speech (blue) and distractors: traffic (pink), TV (magenta), radio (green), and recordings from LDC databases (orange).*

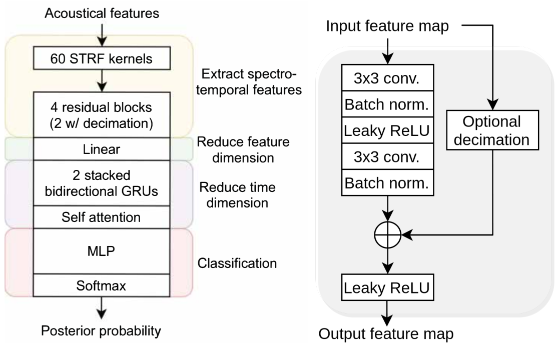

Figure 2: *Block diagram of the proposed system (left) and the zoomed-in view of one residual block (right).*

could potentially discriminate speaking styles in live speech from broadcast speech. Figure 2 describes the overall organization of the STRFNet system based on deep learning principles. After the extraction of acoustic features, the major components of the processing are: (1) a layer of 2-dimensional convolutional kernels where each is re-parameterized as an STRF, (2) a series of convolutional layers with residual connections that further extract spectro-temporal features, (3) a stacked bidirectional gated recurrent unit (GRU) with self-attention that compresses the time dimension, and (4) a series of feed-forward layers which end with a softmax normalization leading to a single probability of live speech. We describe all of these stages in greater detail in the paragraphs below.

Initial signal processing. We considered three front ends for increased flexibility in initial processing: the constant-Q transform (CQT) [14, 15], the log-mel spectrogram, and a bank of Gammatone filters [16] implemented as convolutional kernels where each kernel has a learnable filter bandwidth and multiplicative gain. For the latter, the unit sample responses of the filters are all truncated at 513 samples to speed up training. This option was motivated by the benefits gained by kernelizing filter responses [13, 12] in a CNN. While the choice of CQT was motivated by the later STRF processing in which the spectral modulation template assumes a fixed logarithmic frequency spacing, we found that the mel frequency spacing performed equally well on the tasks we considered. Moreover, we did not observe benefits from having learnable bandwidth and filter gains provided in the Gammatone filterbank. Therefore, we chose the computationally-efficient log-mel spectrogram.

The learnable STRF layer and residual blocks. To extract speech-specific spectro-temporal features in noisy and reverberant environments effectively, we proposed a convolutional layer in which each kernel is constrained to be an STRF [17] that models modulation on a log-frequency/linear-time scale. We used the same seed functions and the parameterization process proposed by Chi *et al.* [17] and defined the learnable upward and downward-drifting discrete STRFs as follows.

$$\begin{aligned}\mathrm{STRF}_{N,K}[n,k;\Omega,\omega,\phi,\theta]_{\Uparrow} &= \mathrm{STRF}(\tfrac{n}{N},\tfrac{k}{K};\Omega,\omega,\phi,\theta)_{\Uparrow}\\ \mathrm{STRF}_{N,K}[n,k;\Omega,\omega,\phi,\theta]_{\Downarrow} &= \mathrm{STRF}(\tfrac{n}{N},\tfrac{k}{K};\Omega,\omega,\phi,\theta)_{\Downarrow}\end{aligned} \qquad (1)$$

where the continuous STRF functions correspond to the definition in [17], N is the frame rate in Hz, and K is the number of frequency channels per octave. During neural network training, the two modulation frequencies (Ω and ω) and two characteristic phases (ϕ and θ), each referring to spectral and temporal processing, respectively, were adapted by gradient backpropagation. This definition requires a finite time and frequency support, respectively; we discuss our own choices below.

In calculating the discrete STRF, the discrete Hilbert transform will be used in place of its continuous-time counterpart. Since the filter that produces the analytic function of an input signal has infinite impulse response, approximations need to be made so that the gradients backpropagate efficiently during training. For this reason, we design a finite impulse response Hilbert transform filter by the simple frequency-sampling method [18, p. 394]. The M-point discrete Fourier transform causes the analytic signal to be time-aliased to some extent; we use $M = 512$ for all our experiments.

To enhance the extracted spectro-temporal features, we follow the STRF convolutional layer with four residual blocks, whose architecture is illustrated in the right figure of Figure 2. This design is motivated by the success of residual connection in image recognition [19] as well as its successful application in spoofing detection [20, 21].

Bidirectional GRUs with self attention. After spectro-temporal feature extraction through the CNNs, features of each time frame are aggregated, flattened, and passed through a fully connected layer for further reduction of dimensionality. Then, we use a stack of two GRUs to learn long temporal patterns. Following the GRUs, we use a self-attentive pooling layer [22, 23] to compress the time dimension. The self attention achieves this by using a trainable layer that assigns a learnable weight to each time frame and then performs a weighted average to obtain a single feature vector.

After temporal modeling and the self-attentive pooling, the feature vector is passed through a one-hidden-layer multilayer perceptron (MLP) with two dimensions for the output. Finally, softmax is applied to obtain the posterior probability of a given segment being Live Speech. An implementation of the STRFNet system can be found at http://www.cs.cmu.edu/~robust/code.html

3. Evaluation Tasks and Datasets

Datasets. We evaluated our proposed system on the VTD task and the spoofing detection tasks in the most recent ASVspoof

2019 challenge [7]. In this section, we describe the data we used as well as our evaluation plan.

As described in the introduction, SRI collected the data specifically for the VTD task. The dataset consists of 34 recording sessions, 20 of which are used for training, 8 are used in development, and 6 are used for evaluation. The training data were recorded in Room 1 and Room 2 and the development data and evaluation data were recorded in Room 3 and Room 4, respectively. Each recording session lasted an average of 9 hours, live speech was only present for an average of 72 minutes scattered throughout the session. The difficulty of this task comes from the fact that live speech was only present for a small fraction of each session and was often time-overlapping with distractor audio such as television, radio broadcasts and non-broadcast-style speech from Linguistic Data Consortium (LDC) databases at low SNRs. The lower panel of Figure 1 summarizes the live speech and distractor audio time regions for the development room along with a schematic diagram of the room. Each of these sessions were recorded from 5 far-field microphones dispersed around the room. During testing, we only have access to one of the microphones at a time. In total, the dataset comprised of roughly 800 hours, 400 hours, and 400 hours of audio recordings in the training data, development data, and evaluation data, respectively, and were all sampled at 11,025 Hz.

Although the STRFNet system was developed for the purpose of VTD, we wanted to test our system and evaluate the benefits of our proposed learnable STRF layer on a similar task that was publicly benchmarked. We chose the ASVspoof 2019 challenge dataset [24] (ASVspoof henceforth) since their tasks, both logical access (LA) and physical access (PA), are similar to VTD. To better match the more challenging acoustical conditions of the VTD task we added VTD distractor audio to the ASVspoof data and downsampled the data to 11,025 Hz.

We found in our pilot experiments that our baseline system performed comparably to reported results obtained from the original data [7] on both the LA and PA tasks. However, the performance for the PA task became much worse after the audio was downsampled and combined with VTD distractor audio. This suggests that the good benchmark performance on the PA task [8] was enabled by subtle spectral distinctions that do not survive decimation and additive noise. For these reasons we evaluate only on the LA task (ASVspoof-LA henceforth). The modified ASVspoof-LA dataset consists of 24 hours, 24 hours, and 63 hours of training, development, and evaluation data, respectively, and 44 hours of VTD distractor audio that are randomly sampled and added.

Evaluation metrics. For the VTD task, the systems were evaluated using the Detection Cost Function (DCF) proposed by the organizer as the primary metric and Equal Error Rate (EER). The proposed DCF is a weighted average of the probability of false alarm (P_{FA}) and probability of a miss (P_M), with misses weighted three times as much as false alarms. Both probabilities are normalized according to the duration of segments that are labeled as Live Speech versus Distractor audio. For the ASVspoof-LA task, we evaluated the systems using the EER at various SNRs. When evaluating on clean speech only, we also used the tandem Detection Cost Function (t-DCF) [25], the primary metric in the AVSspoof 2019 challenge, which combines the performance of both automatic speaker verification (ASV) and spoofing detection. Since the provided ASV score was obtained on clean speech, we did not calculate the t-DCF when evaluating on noisy data.

Table 1: *System configurations for VTD and ASVspoof tasks. Subscript S denotes static STRFs. The remainder of the systems are identical and follow Figure 2.*

System	First layer (VTD/ASVspoof)	# Kernels
CNN$_K$	2DConv	K
Hybrid$_S$	– / 2DConv+STRFConv	– / 60+60
Hybrid	– / 2DConv+STRFConv	– / 60+60
STRFNet$_S$	STRFConv / –	60 / –
STRFNet	STRFConv / –	60 / –

4. Experimental Procedures and Results

In this section, we describe the experimental procedures for both tasks and discuss the results.

Feature front end and data augmentation. As described earlier, we used the log-mel spectrogram as the front end. The spectrogram was obtained using a 20-millisecond Hamming window with 50% overlap between frames and a 512-point discrete Fourier transform. In all our experiments, 40 mel bands were used for the VTD task and 80 were used for the ASVspoof-LA task. Logarithmic power compression using natural logarithm was applied subsequently.

To help our systems generalize to unseen speech and distractor audio, we applied the SpecAugment [26] method to the input feature during training. SpecAugment is a simple data augmentation method that performs time warping and randomly masks entire frequency bands and time frames of the input spectrogram representation before passing it as input to the system. SpecAugment was initially proposed to improve automatic speech recognition [26], but we found it to be helpful for both the VTD and ASVspoof tasks.

Baseline systems. As shown in Table 1, we included five types of baseline systems to study the effectiveness of the learnable STRFs. In the first convolutional layer, CNN$_K$ consists of generic kernels (2DConv); Hybrid systems consist of both generic and STRF kernels (STRFConv); STRFNet consists of only STRF kernels. The conventional CNN$_K$ was tuned to produce competitive results compared to those reported in the official challenge [7]. (The subscript K refers to the number of kernels in the first layer in a given system.) After obtaining satisfactory baseline results on ASVspoof-LA, we matched the number of kernels for the other systems. The STRFs in Hybrid$_S$ and STRFNet$_S$ were initialized with random parameters that were fixed during training. Specifically, the temporal and spectral modulation frequencies are uniformly sampled over $[0, 10)$ Hz and $[0, 0.2)$ cycles per channel, respectively, while characteristic phases are uniformly sampled over $[0, 2\pi)$. For the VTD task, the STRF has a time support of .5 seconds and spans 15 frequency channels. For the ASVspoof-LA task, the STRF has a time support of .25 seconds and spans 7 frequency channels.

Training and evaluation procedure. To train our systems for the VTD task, we first randomly selected 5-second audio segments and computed a log-mel spectrogram. Next, we normalized each channel by subtracting the mean and dividing by the standard deviation of the channel across a single segment. We then applied SpecAugment to the normalized features and used them as input to the systems. During testing, the systems took in one 5-second segment at a time and output raw posterior probability scores for that segment. Before we calculated the DCF, we made a binary decision on each posterior probability score for each segment based on a fixed threshold. We then applied a postprocessing step that re-labels as live speech selected brief segments that had been hypothesized as distrac-

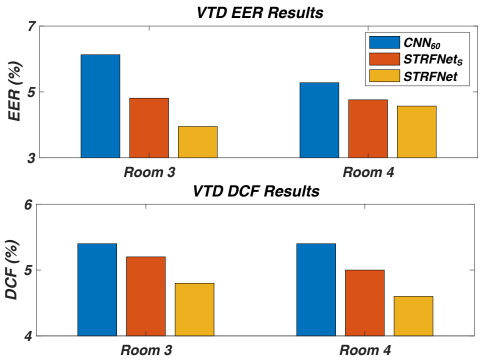

Figure 3: *EER and DCF scores for the VTD task.*

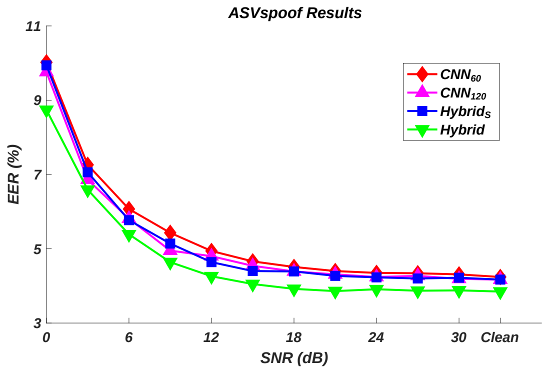

Figure 4: *Equal error rates from ASVspoof-LA task.*

Table 2: *Baseline performance of CNN$_{60}$ on ASVspoof-LA task.*

SpecAug. [26]	Samp. Rate (Hz)	EER (%)	t-DCF
Not Used	16000	6.49	.116
Used	16000	6.02	.091
Not Used	11025	8.20	.187
Used	11025	6.40	.166

tor audio. We used the fixed threshold for each specific system that provided the lowest DCF for that system. To calculate an EER that reflects the post-processing step used in the DCF procedure, we also apply the postprocessing step before the false positive rates and false negative rates are calculated.

To train our systems on the downsampled ASVspoof-LA data, we mixed the speech in every training file with a segment of randomly sampled VTD noise at global SNRs of $5, 10, 15, 20, 25, 30, 40$ dB, selected randomly. We only applied SpecAugment to the input features during training. The VTD distractor audio for evaluation was disjoint from the training data. We evaluated the EER of all systems at more finely sampled SNRs. Both the EER and t-DCF were calculated when evaluating our systems on clean speech.

All systems were trained using the Adam optimizer [27] with a learning rate of 10^{-4} and a batch size of 64 in PyTorch [28]. For both tasks, we stopped training when the performance on the development set stopped improving for multiple epochs. Each system has about 2.4 million trainable parameters.

Baseline results. Table 2 shows ASVspoof-LA results for our baseline system CNN$_{60}$. When trained with SpecAugment using the original data, CNN$_{60}$ produces a t-DCF of .091, which is comparable to that of the 4^{th} place system in the official 2019 challenge for this task [7]. We used the configuration highlighted in gray to train all our systems.

VTD results and discussion. Figure 3 summarizes the performance of the systems on the VTD task. The proposed learnable STRFNet system outperforms both the baseline CNN$_{60}$ system and the STRFNet$_S$ system (which does not adapt) in both Rooms 3 and 4. Our results show that replacing a generic convolutional layer with a static STRF layer reduces both the DCF and EER on evaluation Room 4 by 5.7 % relative and 9.9 % relative, respectively. Furthermore, we can obtain additional improvement by enabling the learnable component in the STRF layers, further reducing the DCF and EER by 8.0 % relative and 4.0 % relative, respectively.

We observed during the experiment that the STRFNet system was able to reject unseen noise with high confidence. However, the television and radio speech remains a challenging distractor for all systems, especially at low SNRs. Selecting suitable STRFs that discriminate different types of speech is an interesting area that we will explore in the future.

AVSspoof results and discussion. The performance of each system on the downsampled ASVspoof-LA data with VTD distractor audio added at various SNRs is shown in Figure 4. Our Hybrid system outperforms the original baseline CNN$_{60}$ by an average of 11.68% relative EER and outperforms the two comparable baseline systems CNN$_{120}$ and Hybrid$_S$ by an average of 8.61% and 8.56% relative EER, respectively. Our results show that our Hybrid model benefits from having both generic convolutional kernels and our proposed learnable STRF kernels in the first layer. During the experiments, we observed that STRFNet did not perform well on this task. Nevertheless, the Hybrid system both performed the best and was the most robust to unseen noise and synthesis methods. This suggests that the STRFs effectively reject distractor noise, but are by themselves not sufficient for discriminating real from synthetic speech.

In general, we find that degrading speech is an effective training technique that improves the systems generalization capabilities to the unseen spoofing techniques in the evaluation data. As shown in Table 2, SpecAugment drastically improved the system trained on the downsampled data. In addition, adding VTD distractor audio to the downsampled data during training further improved the EER of all the systems when evaluating on all conditions, including clean speech.

5. Conclusions

In this paper, we incorporate learnable spectro-temporal receptive fields (STRFs) in a deep neural network for the emerging task of voice type discrimination (VTD). We show that systems using the proposed learnable STRFs in the first layer consistently outperform a competitive baseline using generic kernels for the VTD task and for the logical access task in the ASVspoof 2019 challenge [7]. We also show that the learning component of the STRF kernel is essential for both robust spoofing detection at a wide range of unseen noisy environments and the VTD task. In the future, we plan on comparing the Gabor-based implementation of the STRF kernels [29] to the implementation used in this paper [17]. We will also evaluate the robustness of the learnable STRFs for other tasks such as robust speech recognition and robust speaker identification.

6. Acknowledgement

This work was funded by the Johns Hopkins University Applied Physics Laboratory.

7. References

[1] S. Shiota, F. Villavicencio, J. Yamagishi, N. Ono, I. Echizen, and T. Matsui, "Voice liveness detection for speaker verification based on a tandem single/double-channel pop noise detector." in *Odyssey*, 2016, pp. 259–263.

[2] M. Sahidullah, D. A. L. Thomsen, R. G. Hautamäki, T. Kinnunen, Z.-H. Tan, R. Parts, and M. Pitkänen, "Robust voice liveness detection and speaker verification using throat microphones," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 44–56, 2017.

[3] M. Wu, S. Panchapagesan, M. Sun, J. Gu, R. Thomas, S. N. P. Vitaladevuni, B. Hoffmeister, and A. Mandal, "Monophone-based background modeling for two-stage on-device wake word detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5494–5498.

[4] J. Li, S. Qu, X. Li, J. Szurley, J. Z. Kolter, and F. Metze, "Adversarial music: Real world audio adversary against wake-word detection system," in *Advances in Neural Information Processing Systems*, 2019, pp. 11 908–11 918.

[5] C. Richey, Z. Armstrong, and A. Lawson, "Distant microphone conversational speech in noisy environments," SRI International, Tech. Rep., 2019.

[6] Y. Kida and T. Kawahara, "Evaluation of voice activity detection by combining multiple features with weight adaptation," in *INTERSPEECH*, 2006.

[7] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. H. Kinnunen, and K. A. Lee, "ASVspoof 2019: Future horizons in spoofed and fake audio detection," in *INTERSPEECH*, 2019, pp. 1008–1012.

[8] L. Li, Y. Chen, D. Wang, and T. F. Zheng, "A study on replay attack and anti-spoofing for automatic speaker verification," in *INTERSPEECH*, 2017, pp. 92–96.

[9] K. Wang and S. A. Shamma, "Spectral shape analysis in the central auditory system," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 382–395, 1995.

[10] N. Mesgarani, M. Slaney, and S. A. Shamma, "Discrimination of speech from nonspeech based on multiscale spectro-temporal modulations," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 3, pp. 920–930, 2006.

[11] P. Agrawal and S. Ganapathy, "Modulation filter learning using deep variational networks for robust speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 244–253, 2019.

[12] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with SincNet," in *IEEE Workshop on Spoken Language Technology*, 2017, pp. 1021–1028.

[13] E. Loweimi, B. P, and S. Renals, "On learning interpretable CNNs with parametric modulated kernel-based filters," in *INTERSPEECH*, 2019, pp. 3480–3484.

[14] J. C. Brown, "Calculation of a constant Q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 424–434, 1991.

[15] J. C. Brown and M. S. Puckette, "An efficient algorithm for the calculation of a constant Q transform," *The Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2698—2701, 1992.

[16] R. D. Patterson, M. H. Allerhand, and C. Giguere, "Time-domain modeling of peripheral auditory processing: a modular architecture and a software platform," *The Journal of the Acoustical Society of America*, vol. 98, no. 4, pp. 1980–1984, 1995.

[17] T. Chi, P. Ru, and S. A. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 887–906, 2005.

[18] J. S. Lim and A. V. Oppenheim, *Advanced topics in signal processing*. Prentice-Hall, Inc., 1987.

[19] 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.

[20] W. Cai, H. Wu, D. Cai, and M. Li, "The DKU replay detection system for the ASVspoof 2019 challenge: On data augmentation, feature representation, classification, and fusion," in *INTERSPEECH*, 2019, pp. 1023–1027.

[21] M. Alzantot, Z. Wang, and M. B. Srivastava, "Deep residual neural networks for audio spoofing detection," in *INTERSPEECH*, 2019, pp. 1078–1082.

[22] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Odyssey*, 2018, pp. 74–81.

[23] M. India, P. Safari, and J. L. Hernando, "Self multi-head attention for speaker recognition," in *INTERSPEECH*, 2019, pp. 4305–4309.

[24] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee, L. Juvela, P. Alku, Y.-H. Peng, H.-T. Hwang, Y. Tsao, H.-M. Wang, S. L. Maguer, M. Becker, F. Henderson, R. Clark, Y. Zhang, Q. Wang, Y. Jia, K. Onuma, K. Mushika, T. Kaneda, Y. Jiang, L.-J. Liu, Y.-C. Wu, W.-C. Huang, T. Toda, K. Tanaka, H. Kameoka, I. Steiner, D. Matrouf, J.-F. Bonastre, A. Govender, S. Ronanki, J.-X. Zhang, and Z.-H. Ling, "ASVspoof 2019: a large-scale public database of synthetic, converted and replayed speech," *Computer Speech and Language*, vol. 64, p. 101114, 2020.

[25] T. Kinnunen, K.-A. Lee, H. Delgado, N. W. D. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "t-DCF: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification," in *Odyssey*, 2018, pp. 312–319.

[26] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple augmentation method for automatic speech recognition," in *INTERSPEECH*, 2019, pp. 2613–2617.

[27] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 2014.

[28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*, 2019, pp. 8026–8037.

[29] B. T. Meyer and B. Kollmeier, "Robustness of spectro-temporal features against intrinsic and extrinsic variations in automatic speech recognition," *Speech Communication*, vol. 53, no. 5, pp. 753–767, 2011.