END-TO-END ANTI-SPOOFING WITH RAWNET2

Hemlata Tak[1]*, Jose Patino*[1]*,Massimiliano Todisco*[1]*, Andreas Nautsch*[1]*,*
Nicholas Evans[1] *and Anthony Larcher*[2]

[1]EURECOM, Sophia Antipolis, France
[2]LIUM - Université du Maine, Le Mans, France

ABSTRACT

Spoofing countermeasures aim to protect automatic speaker verification systems from attempts to manipulate their reliability with the use of spoofed speech signals. While results from the most recent ASVspoof 2019 evaluation show great potential to detect most forms of attack, some continue to evade detection. This paper reports the first application of RawNet2 to anti-spoofing. RawNet2 ingests raw audio and has potential to learn cues that are not detectable using more traditional countermeasure solutions. We describe modifications made to the original RawNet2 architecture so that it can be applied to anti-spoofing. For A17 attacks, our RawNet2 systems results are the second-best reported, while the fusion of RawNet2 and baseline countermeasures gives the second-best results reported for the full ASVspoof 2019 logical access condition. Our results are reproducible with open source software.

***Index Terms*—** anti-spoofing, countermeasures, presentation attack detection, automatic speaker verification

1. INTRODUCTION

Our recent efforts to improve the reliability of spoofing countermeasures for automatic speaker verification (ASV) have confirmed that: (i) artefacts from converted voice or synthetic speech spoofing attacks reside at the sub-band level [1]; (ii) detection performance can be improved through the use of high-spectral-resolution front-ends, particularly those which emphasize information within the same sub-bands [2]. When used even with a trivial Gaussian mixture model (GMM) back-end, high-spectral-resolution front-ends lead to competitive performance that would have obtained fourth position (out of 48 submissions) in the ASVspoof 2019 challenge.

Unfortunately, though, some spoofing attacks continue to evade detection, particularly the infamous A17 attack, a neural network voice conversion (VC) based attack that uses a generalised direct waveform modification method for waveform generation. While we have seen that even A17 attacks can be detected when classifiers are learned with representative data (i.e. through learning with evaluation data), performance is poor when the same classifier is learned using only standard training data. Hence, while high-spectral-resolution front-ends *do* capture A17 artefacts, they *cannot* be detected.

The same observation shows that, once training data for a newly discovered spoofing attack is available, then vulnerabilities can be fixed easily simply by re-learning with representative training data. More proactive or preemptive strategies that offer some protection against the unexpected are preferable. This idea embodies the objectives of the ASVspoof initiative since its inception, namely the pursuit of generalisable spoofing countermeasures that are capable of detecting attacks not seen in training data. The latter are characteristic of in-the-wild settings in which the nature of spoofing attacks cannot be predicted and will evolve continuously. The question addressed in this paper is whether A17 attacks can be detected even in the *absence* of representative training data.

This is a difficult question to answer reliably in a post-evaluation setting where the characteristics of attack A17 are now known and cannot be forgotten. One-class classifiers trained only on bona fide data without the use of any spoofed data are an obvious candidate solution and have shown some previous success in anomaly, novelty and spoofing detection tasks [3, 4, 5, 6]. Anecdotal evidence, however, shows that many, including the authors of the current article, have failed to apply them successfully to ASVspoof 2019 data. Since A17 attacks cannot be detected using our current countermeasure (at least not without representative A17 training data) we must seek cues elsewhere.

Our assumption is that the use of hand-crafted features does not offer the best potential to detect unforeseen attacks since they rely too much upon the characterisation of artefacts corresponding to known attacks. Attack-specific artefacts are likely to be insufficient for the detection of unseen attacks and a higher-level, more generalisable representation is needed to ensure robust performance. This paper reports what is, to the best of our knowledge, the first application of RawNet2 [7] deep neural network classifiers to anti-spoofing. RawNet2 operates directly upon the raw speech waveform, circumventing almost entirely any dependence upon hand-crafted features.

This work is partly supported by the ExTENSoR project funded by the French Agence Nationale de la Recherche, ANR-19-CE23-0001-01.

The goal of this work is not just to design more reliable spoofing countermeasures, but also to determine whether such end-to-end approaches that utilise some degree of automatic feature learning can improve performance in a *worst-case scenario* such as that of A17.

The paper is organised as follows. The original RawNet2 architecture and its application to ASV are described in Section 2. Modifications made to RawNet2 in order to apply it successfully to anti-spoofing are described in Section 3. Experimental work is described in Section 4. Our conclusions are presented in Section 5.

2. PREVIOUS WORK

This section provides a brief review of past work that led to the development of RawNet2 [7]. The treatment is upon automatic speaker verification. As in many fields of speech processing, recent years have seen the adoption of end-to-end classifier architectures where every component between, and including any front-end processing and back-end classification, is automatically and jointly optimised. Many of these solutions operate directly upon the raw speech waveform and avoid limitations introduced from the use of knowledge-based, hand-crafted acoustic features. One advantage is the use of representations optimised for the application, rather than the use of generic, fixed-bandwidth decompositions such as those stemming from Fourier based analysis [8].

There is already a sizeable body of work in the automatic speaker verification literature which reports end-to-end solutions that operate on the raw speech waveform. Among other early solutions is RawNet [9], introduced in 2019. RawNet is a convolutional neural network architecture which outputs speaker embeddings. The first convolutional layer is applied directly to the raw speech waveform, with all filter parameters being learned automatically. Among the higher layers are residual blocks [10] which extract frame-level representations. Residual blocks use skip connections that enable the training of deeper classifiers to leverage more discriminative information. They use either long short-term memory (LSTM) as in [11] or gated recurrent units (GRUs) as in [9] to aggregate utterance-level representations and either a b-vector classifier [12, 13] as in [11] or a DNN backend with concatenation and multiplication (concat & mul) technique as in [9] for verification. The use of a wholly unconstrained first layer whose parameters are learned automatically can result in slow learning. The first layer outputs also tend to be noisy, especially when training data is sparse.

One solution to these issues is SincNet [14, 15]. Whereas the higher layers of the SincNet architecture are relatively standard, the first convolutional layer operates upon the raw waveform and consists in a bank of band-pass filters parametrized in the form of sinc functions. The use of a constrained first layer, with fewer learnable parameters whereby only the cut-in and cut-off frequencies are learned with a fixed rectangular-shaped filter response, leads to the learning of a more meaningful filterbank structure and outputs.

RawNet2 [7], proposed in 2020, combines the merits of the original RawNet approach (RawNet1) with those of SincNet. The first layer of RawNet2 is essentially the same as that of SincNet, whereas the upper layers consist of the same residual blocks and GRU layer as RawNet1. New to RawNet2 is the application of filter-wise feature map scaling (FMS) using a sigmoid function applied to residual block outputs as in [16]. FMS acts as an attention mechanism and has the goal of deriving more discriminative representations. The embedding dimension for RawNet2 is also greatly increased, from 128 for RawNet1 to 1024 for RawNet2. Last, whereas RawNet1 obtains better results with a DNN-based back-end classifier, RawNet2 gives better results with a cosine similarity score.

Experiments using the VoxCeleb1 database show that RawNet1 gives an 11% relative improvement in terms of an equal error rate (EER) metric compared to an i-vector based system [17], and is also shown to be competitive with those of an x-vector system [18]. When trained using the VoxCeleb2 database, RawNet2 gives a 20% relative improvement compared to an x-vector based system [19] whereas, using the expanded VoxCeleb1-E evaluation dataset protocol, it gives a 13% relative improvement compared to a Thin-ResNet-34 based system [20]. There is hence plentiful evidence that end-to-end architectures which avoid the use of knowledge-based, hand-crafted features have potential to improve automatic speaker verification performance. The goal of the work reported in this paper is to determine whether the benefit translates also to anti-spoofing, especially in worst case scenarios.

3. APPLICATION TO ANTI-SPOOFING

This section describes modifications made to the original RawNet2 architecture [7] so that it can be applied successfully to anti-spoofing. Modifications to parameters or architecture components are highlighted in Table 1 in bold text.

The first modification concerns the first layer of the architecture which ingests raw speech. Since it leads to worse performance, we did not apply layer normalization [21] to the input. On account of training data sparsity, or rather the limited number of different spoofing attacks (only 6 for the training & development partitions of the ASVspoof 2019 LA database), we neither learn automatically the bandwidth nor spectral position of each sinc filter. Other experiments not reported here show that it leads to overfitting. While SincNet is initialised with a Mel-distributed filterbank, we experimented also with linearly-distributed and inverse-Mel scaled filterbank. These choices are motivated by the success of linear and inverse-Mel [22] front-ends for anti-spoofing. We fixed the duration of the raw waveform input to $\approx$ 4 sec (64000 samples) by either cropping long utterances or concatenating short utter-

Table 1. The RawNet2 architecture used for anti-spoofing. Modifications made to the original architecture are highlighted in boldface. BN refers to batch normalisation.

Layer	Input: **64000** samples	Output shape
Fixed Sinc filters	Conv(**129**,1,128) Maxpooling(3) BN & LeakyReLU	(**21290**,128)
Res block	BN & LeakyReLU Conv(3,1,128) { BN & LeakyReLU Conv(3,1,128) } ×2 Maxpooling(3) FMS	(**2365**,128)
Res block	BN & LeakyReLU Conv(3,1, **512**) { BN & LeakyReLU Conv(3,1, **512**) } ×4 Maxpooling(3) FMS	(**29**,512)
GRU	GRU(1024)	(1024)
FC	1024	(1024)
Output	**1024**	**2**

ances such that all utterances have the same length.

In similar fashion to [7], we also optimised the filter length (impulse response duration). This is because the duration of cues used to detect spoofing are not necessarily the same as those for speaker recognition. Whereas the original work used filter lengths of 251 samples, we observed better results using only 129 samples. Whereas the configuration of the first residual block is left unchanged, we use a larger number of kernel filters (512) in second residual block. As for RawNet2, we apply FMS independently to the output of each residual-block, thereby helping to emphasize the most informative filter outputs. We adopt the combined additive and multiplicative feature scaling approach in [7].

A GRU layer with 1024 hidden nodes is applied to aggregate frame-level representations into a single utterance-level representation. Rather than producing embeddings as in RawNet 2, the GRU output is followed by an additional fully connected layer which precedes the output layer. A softmax activation function is then applied in order to produce two-class predictions: bona-fide or spoof. The network was trained with ADAM optimisation using a learning rate of 0.0001, 100 epochs and a mini-batch size of 32. The full architecture is summarised in Table 1.

4. EXPERIMENTAL WORK

The following describes the ASVspoof 2019 LA database, evaluation metrics, baseline countermeasure and results. We report results for three different RawNet2 variants and results for fused systems.

4.1. ASVspoof 2019 and t-DCF metric

The ASVspoof 2019 logical access (LA) database consists of three independent partitions: train, development and evaluation. Spoofed attacks are generated using a set of 19 different speech synthesis, voice conversion and hybrid algorithms [23]. The training and development partitions contain bona fide and spoofed data generated with 6 different attack algorithms (A01-A06), whereas the evaluation partitions contains bona fide speech and spoofed data generated with 13 different algorithms (A07-A19). As permitted by ASVspoof rules [24], we re-partitioned the data by supplementing training data with 90% of development data. The remaining 10% was used for validation. The primary metric is the minimum normalised tandem detection cost function (t-DCF) metric [25, 26]. For consistency with results reported in [27], t-DCF results derived according to the original min t-DCF metric in [25, 26]. Results are also reported in terms of the pooled equal error rate (EER) computed using [28].

4.2. Baseline

The baseline classifier for this work is the high-spectral resolution linear frequency cepstral coefficient (LFCC) countermeasure reported in [2]. It uses 70 linearly-spaced filters with conventional cepstral analysis and a GMM back-end classifier. Despite its simplicity, it outperforms all but three systems submitted to the ASVspoof 2019 challenge.

4.3. RawNet2 results

Table 2 shows results in terms of min t-DCF for the baseline (L) and three different RawNet configurations (S1-3). Attacks A01-06 correspond to development data whereas A07-A19 correspond to evaluation data. P1 and P2 columns show pooled min t-DCF and EER results respectively for each partition.

Results for all three RawNet classifiers are inferior to those of the baseline. Focusing only upon the evaluation set, the pooled min t-DCF for the baseline is 0.09, whereas the best performing S2 RawNet2 system which uses fixed inverse Mel-scaled sinc filters gives a min t-DCF of 0.1175. Results for the infamous A17 attack, i.e. the worst case for the baseline and all of the top-performing ASVspoof 2019 LA submissions, are more favourable. Whereas the baseline classifier gives a min t-DCF of 0.3524 for A17, the S3 RawNet2 classifier that uses fixed, linearly-scaled sinc filters gives a min t-DCF of 0.181. To the best of our knowledge, this is among very best published results for A17.

4.4. Fusion

Since the LFCC-GMM baseline system gives the best over-all pooled result, whereas the S3 RawNet system performs best

Table 2. Results in terms of min t-DCF for development (A01-A06) and evaluation (A07-A19) partitions and respective pooled min t-DCF (P1) and pooled EER (P2). L: High-spectral-resolution LFCC. S1: fixed Mel-scaled sinc filters. S2: fixed inverse Mel-scaled sinc filters. S3: fixed linear-scale sinc filters.

S	A01	A02	A03	A04	A05	A06	P1	P2	A07	A08	A09	A10	A11	A12	A13	A14	A15	A16	A17	A18	A19	P1	P2
L	.0000	.0000	.0000	.0005	.0000	.0000	.0000	.00	.0011	.0009	.0003	.1536	.0048	.1150	.0798	.0697	.0695	.0007	.3524	.0741	.0084	**.0904**	**3.50**
S1	.0443	.0125	.0274	.0435	.0598	.0653	.0460	1.36	.0277	.1984	.0118	.0373	.0195	.0379	.0433	.0192	.0314	.0457	.2620	.6145	.0526	.1301	5.64
S2	.0269	.0025	.0045	.0291	.0297	.0347	.0285	0.86	.0131	.0551	.0131	.0300	.0100	.0360	.0076	.0046	.0344	.0187	.2626	.6237	.0409	.1175	5.13
S3	.0230	.0315	.0117	.0468	.0354	.0690	.0400	1.25	.0382	.1106	.0348	.0493	.0505	.0445	.0459	.0380	.0378	.0460	**.1810**	.5285	.0679	.1294	4.66

Table 3. Performance for the ASVspoof 2019 evaluation partition in terms of pooled min t-DCF and pooled EER for top-performing systems (T05, T45, T60 and T24), SVM-based fusions of high-spectral-resolution LFCC (L) [2] and ResNet2 systems (boldface), and ASVspoof 2019 baseline systems (B1, B2). Individual min t-DCF results for A17 are illustrated in the right-most column.

System	min-tDCF	EER	min t-DCF (A17)
T05	0.0069	0.22	0.0040
L+S1	**0.0330**	**1.12**	**0.1161**
L+S1+S2+S3	**0.0347**	**1.14**	**0.0808**
L+S3	**0.0370**	**1.14**	**0.0965**
L+S2	**0.0443**	**1.35**	**0.1339**
T45 [29]	0.0510	1.86	0.2208
T60 [30]	0.0755	2.64	0.3254
L [2]	0.0904	3.50	0.3524
T24	0.0953	3.45	0.3547
LFCC:B2 [27]	0.2116	8.09	0.2042
CQCC:B1 [27]	0.2366	9.57	0.5859

for the worst case A17 attack, it is of interest to evaluate the benefit of their fusion.

Experiments were conducted using the support vector machine (SVM) based fusion approach described in [2]. It was used to fuse the high-spectral-resolution countermeasure (L) with S1, S2 and S3 RawNet2 systems. Fusion results are illustrated in Table 3, with the top-4 ASVspoof 2019 challenge results. Also shown are results for the two official ASVspoof 2019 baseline systems: the CQCC B1 baseline and the low-spectral-resolution LFCC B2 baseline.

All systems outperform the official baselines, with the high-spectral-resolution baseline (L) giving a min t-DCF of 0.0904, marginally better than the min t-DCF result of 0.0953 for team T24. Team T60 produced a min t-DCF of 0.0755 and team T45 a result of 0.0510. All fusions of the high-spectral-resolution countermeasure (L) with either S1, S2 and S3 RawNet2 systems perform better, with the L+S1 combination even outperforming fusion with all three RawNet systems (L+S1+S2+S3).

The last column of Table 3 shows performance in terms of min t-DCF for each countermeasure system for attack A17. Just like individual system results shown in Table 2, all RawNet2 fused systems produce a substantially lower min t-DCF than the baseline system (L), confirming that the end-to-end RawNet2 system is complementary; it is learning artefacts that the baseline system is not. Last, while the results for team T05 show a lower pooled min t-DCF and lower min t-DCF for attack A17, our results are reproducible with open source software[1].

5. DISCUSSION AND CONCLUSIONS

This paper reports the first successful application of RawNet2 to automatic speaker verification anti-spoofing. While pooled results are inferior to those of the baseline countermeasure, those for the infamous, worst case A17 voice conversion spoofing attack are the second best reported thus far. Whether the comparison of countermeasure solutions should be based upon pooled or worst case results is an interesting question since, once a vulnerability is exposed, then attacks that exploit it will likely occur more frequently than less effective attacks. Despite the competitiveness of our proposed solution, the min t-DCF for attack A17 is still over 250% higher than for the pooled min t-DCF.

The fusion of baseline and RawNet2 classifier scores give the second best results reported in the literature. Fusion results suggest that the RawNet2 classifier is learning cues that are different to those learned by the baseline classifier. This observation raises the questions of what these cues are and what about the RawNet2 classifier enables it to learn them. Our current hypothesis is that, aside from operating upon the raw signal, the ability comes from temporal attention, an ability which the baseline classifier clearly lacks. This hypothesis would seem to be supported by the presence of occasional, punctual clicking noises that characterise A17 attacks. Our suspicion is that A17 artefacts are phase-related and that they are captured by the linear-phase filters in the first layer of the RawNet2 architecture which hence produce phase-aligned waveforms to the first residual block. The validation of this hypothesis is the subject of ongoing work. Other directions for future work concern the exploration of embeddings and alternative back-end approaches to classification.

[1] URL to be made available upon publication.

6. REFERENCES

[1] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "An explainability study of the constant Q cepstral coefficient spoofing countermeasure for automatic speaker verification," in *Proc. Speaker Odyssey Workshop*, Tokyo, Japan, 2020, pp. 333–340.

[2] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "Spoofing Attack Detection using the Non-linear Fusion of Sub-band Classifiers," *Proc. INTERSPEECH*, 2020.

[3] F. Alegre, A. Amehraye, and N. Evans, "A one-class classification approach to generalised speaker verification spoofing countermeasures using local binary patterns," in *Proc. BTAS*, 2013, pp. 1–8.

[4] S. Fatemifar, S. R. Arashloo, M. Awais, and J. Kittler, "Spoofing attack detection by anomaly detection," in *Proc. ICASSP*, 2019, pp. 8464–8468.

[5] S. Fatemifar, M. Awais, S. R. Arashloo, and J. Kittler, "Combining multiple one-class classifiers for anomaly based face spoofing attack detection," in *Proc. ICB*, 2019, pp. 1–7.

[6] T. Ohki, V. Gupta, and M. Nishigaki, "Efficient Spoofing Attack Detection against Unknown Sample using End-to-End Anomaly Detection," in *Proc. APSIPA ASC*, 2019, pp. 224–230.

[7] J. Jung, S. Kim, H. Shim, J. Kim, and H. Yu, "Improved RawNet with Filter-wise Rescaling for Text-independent Speaker Verification using Raw Waveforms," in *Proc. INTERSPEECH*, 2020.

[8] H. Muckenhirn, M. M. Doss, and S. Marcell, "Towards directly modeling raw speech signal for speaker verification using CNNs," in *Proc. ICASSP*, 2018, pp. 4884–4888.

[9] J. Jung, H. Heo, J. Kim, H. Shim, and H. Yu, "Rawnet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification," in *Proc. INTERSPEECH*, 2019, pp. 1268–1272.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. ICVPR*, 2016, pp. 770–778.

[11] J. Jung, H. Heo, I. Yang, H. Shim, and H. Yu, "A complete end-to-end speaker verification system using deep neural networks: From raw signals to verification result," in *Proc. ICASSP*, 2018, pp. 5349–5353.

[12] H. Lee, Y. Tso, Y. Chang, H. Wang, and S. Jeng, "Speaker verification using kernel-based binary classifiers with binary operation derived features," in *Proc. ICASSP*, 2014, pp. 1660–1664.

[13] H. Heo, I. Yang, M. Kim, S. Yoon, and H. Yu, "Advanced b-vector system based deep neural network as classifier for speaker verification," in *Proc. ICASSP*, 2016, pp. 5465–5469.

[14] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *Proc. Spoken Language Technology Workshop (SLT)*, 2018, pp. 1021–1028.

[15] M. Ravanelli and Y. Bengio, "Learning speaker representations with mutual information," in *Proc. INTERSPEECH*, 2019, pp. 1153–1157.

[16] S. Woo, J. Park, J. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Proc. the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[17] S. Shon, H. Tang, and J. Glass, "Frame-level speaker embeddings for text-independent speaker recognition and analysis of end-to-end model," in *Proc. SLT*, 2018, pp. 1007–1013.

[18] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," in *Proc. INTERSPEECH*, 2018, pp. 2252–2256.

[19] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. ICASSP*, 2018, pp. 5329–5333.

[20] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "VoxCeleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, 2020, 101027.

[21] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[22] M. Sahidullah, T. Kinnunen, and C. Hanilci, "A comparison of features for synthetic speech detection," in *Proc. INTERSPEECH*, pp. 2087–2091.

[23] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A Lee, et al., "ASVspoof 2019: a large-scale public database of synthetized, converted and replayed speech," *Computer Speech & Language (CSL)*, vol. 64, 2020, 101114.

[24] "ASVspoof 2019: the Automatic Speaker Verification Spoofing and Countermeasures Challenge Evaluation Plan," `http://www.asvspoof.org/asvspoof2019/asvspoof2019_evaluation_plan.pdf`.

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

[26] T. Kinnunen, H. Delgado, N. Evans, K. A. Lee, V. Vestman, A. Nautsch, M. Todisco, X. Wang, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "Tandem assessment of spoofing countermeasures and automatic speaker verification: Fundamentals," *IEEE/ACM Transactions on Audio Speech and Language Processing (TASLP)*, vol. 28, pp. 2195 – 2210, 2020.

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

[28] N. Brümmer and E. de Villiers, "The BOSARIS toolkit: Theory, algorithms and code for surviving the new DCF," *arXiv preprint arXiv:1304.2865*, 2013.

[29] G. Lavrentyeva, S. Novoselov, A. Tseren, et al., "STC antispoofing systems for the ASVspoof2019 challenge," in *Proc. INTERSPEECH*, 2019, pp. 1033–1037.

[30] B. Chettri, D. Stoller, V. Morfi, M. A. M. Ramírez, E. Benetos, and B. L Sturm, "Ensemble models for spoofing detection in automatic speaker verification," in *Proc. INTERSPEECH*, 2019, pp. 1118–1112.