2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge
16 September 2021, Online

Investigation on activation functions for robust end-to-end spoofing attack detection system

Woo Hyun Kang, Jahangir Alam*, Abderrahim Fathan*

Computer Research Institute of Montreal (CRIM)
woohyun.kang,jahangir.alam,abderrahim.fathan@crim.ca

Abstract

The main objective of the spoofing attack detection system is to detect the artifacts caused by the spoof generation process (i.e., speech synthesis or voice conversion) given a speech sample. Since the selection of an activation function can affect the ability of the end-to-end spoof detection to focus on the relevant regions of the feature map, we investigate the effects of different activation functions in the antispoofing countermeasure task. From our results, it could be found that different activation functions enable the end-to-end system to learn complementary information for spoof detection. In order to exploit the complementarity between various activation functions, we propose to adopt the activation ensemble technique within the end-to-end system, where the outputs of different activation functions are pooled together. The proposed framework was experimented on the logical access (LA) task ASVSpoof2019 dataset and outperformed the systems using a single activation function.

1. Introduction

The main objective for building a reliable spoofing countermeasure system is to detect the artifacts from the given speech, which may have been caused by the generation process of the spoof attacks. To achieve this, many attempts were made to exploit the techniques which have shown stable performance in the speaker recognition task. In the case of logical access (LA) spoofing detection task the most effective countermeasures are the frame-level acoustic features typically extracted at 10 ms intervals and designed to detect artifacts in the spoofed speech. Classically, the standard Gaussian Mixture Model (GMM) classifier in combination with frame-level acoustic [1–7] or deep features [8] was the most widely adopted spoofing detection approach [1,2,9,10]. But the recent trend in voice anti-spoofing is to employ deep learning architectures in an end-to-end manner on the top of raw signal or hand-crafted acoustic features to discriminate between bonafide and spoof speech signals [9,11–18]. In [11], one class softmax loss with ResNet18 architecture was proposed and outperformed the classical GMM-based methods. In order to further improve the generalization of anti-spoofing systems to unseen test data, several variants of softmax loss were also adopted [11, 15]. Although the conventional end-to-end systems were able to outperform the GMM-based spoof detection systems, they follow the same configuration with the ones used in image classification, which may not be an optimal choice for spoof detection task.

In recent years, as the selection of an activation function can affect the ability of the neural network to extract relevant information from the input data, various variants of the conventional rectified linear unit (ReLU) functions were proposed. Especially in [19], a trainable attention-based activation was introduced to efficiently focus on the relevant regions of the feature map. In light of this, in this paper, we investigate the effect of various activation functions on the performance of the end-to-end spoof detection system. Moreover, in order to exploit the complementary information propagated through different activation functions, we propose to employ the activation ensemble technique to the end-to-end system.

To evaluate the performance of the proposed scheme, we conducted a set of experiments using the ASVSpoof 2019 dataset. The experimental results show that the end-to-end spoof detection can greatly benefit from the usage of multiple activation functions and activation ensembling, outperforming the conventional methods.

The contributions of this paper are as follows:

- We analyze the performance behaviour of anti-spoofing countermeasure systems with different activation functions.
- We compare the countermeasure performance of the proposed activation ensembled system and the conventional methods.

The rest of this paper is organized as follows: The detailed description of end-to-end spoof detection system and conventional activation functions are described in Section 2 and Section 3, respectively. Section 4 describes the activation ensemble technique. The detailed setting for the experimentation and the results are presented in Section 5, and Section 6 concludes the paper.

2. End-to-end anti-spoofing system

Most deep learning based spoofing detection systems employ deep neural architectures, such as residual network (ResNet), on top of hand-crafted/learned features for capturing more discriminative local descriptors which are then aggregated to generate final fixed dimensional utterance-level embeddings. The embeddings are then fed into a classifier which discriminates whether the input audio is a spoof attack or genuine. Conventionally, a two-stage approach was popularly adopted, where the classifier (e.g., support vector machine (SVM)) and the embedding extraction network are trained separately. Recently, in order to mutually optimize the decision hyperplane and the embedding feature space, various end-to-end approaches [9, 11–18] were proposed in the past few years, where the neural classifier is trained jointly with the embedding extraction network.

The proposed model also adopts the end-to-end framework for antispoofing, which is composed of 2 networks: an embedding network and a classification network. For the embedding network, we experimented with the squeeze-and-excitation

*These authors contributed equally to this work

10.21437/ASVSPOOF.2021-13

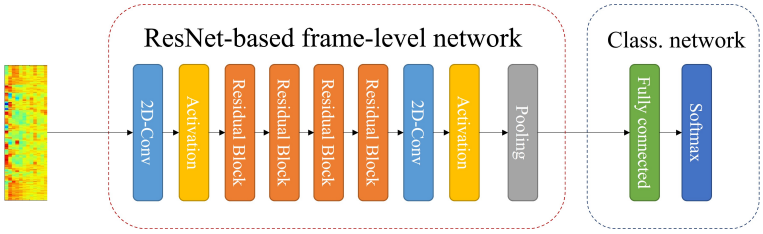

Figure 1: The general architecture of the end-to-end antispoofing countermeasure system. For the residual blocks (orange blocks), ReLU function was used for all activations. The ReLU variants were used after the first and last convolutional layer (yellow blocks).

Table 1: The weight configuration of each layer in the SE-ResNet-18 end-to-end antispoofing system. In this table, ResBlock indicates the Residual Block component in Fig. 1 and L is the length of the input LFCC sequence.

Layer	SE-ResNet-18	Output
Input	-	$1 \times 60 \times L$
2D-Conv	$9 \times 9, 16, stride(3,1)$	$16 \times 18 \times L$
ResBlock	$\begin{bmatrix} \text{2D-Conv } 3 \times 3, 64 \\ \text{2D-Conv } 3 \times 3, 64 \\ \text{FC } 64 \times 4 \\ \text{FC } 4 \times 64 \end{bmatrix} \times 2, stride1$	$64 \times 18 \times L$
ResBlock	$\begin{bmatrix} \text{2D-Conv } 3 \times 3, 128 \\ \text{2D-Conv } 3 \times 3, 128 \\ \text{FC } 128 \times 8 \\ \text{FC } 8 \times 128 \end{bmatrix} \times 2, stride2$	$128 \times 9 \times \frac{L}{2}$
ResBlock	$\begin{bmatrix} \text{2D-Conv } 3 \times 3, 256 \\ \text{2D-Conv } 3 \times 3, 256 \\ \text{FC } 256 \times 16 \\ \text{FC } 16 \times 256 \end{bmatrix} \times 2, stride2$	$256 \times 5 \times \frac{L}{4}$
ResBlock	$\begin{bmatrix} \text{2D-Conv } 3 \times 3, 512 \\ \text{2D-Conv } 3 \times 3, 512 \\ \text{FC } 512 \times 32 \\ \text{FC } 32 \times 512 \end{bmatrix} \times 2, stride2$	$512 \times 3 \times \frac{L}{8}$
2D-Conv	$3 \times 3, 256, stride1$	$256 \times 1 \times \frac{L}{8}$
Pooling	attentive statistics pooling	512
FC	512×256	256
Softmax	256×2	2

residual network (SE-ResNet), which have shown competitive performance in the speaker verificaiton and image classification tasks. Unlike the standard ResNet, a squeeze-and-excitation (SE) block [20] is applied at the end of each non-identity branch of residual block to significantly decrease the computational cost of the system. More specifically we used the SE-ResNet-18, which is an 18 layers deep convolutional network composed of 4 residual blocks. More detailed information of this network architecture is depicted in Table 1.

To aggregate the frame-level output of the ResNet, an attention pooling layer is incorporated where the weighted first and second order (i.e., standard deviation) moments are pooled together across the temporal dimension [9,11,15,16,18] to obtain a utterance-level representation. The pooled statistics are then fed into a neural classifier, which consists of a fully-connected layer and a 2-dimensional softmax layer, where each softmax node represents the bona fide and spoofing classes, respectively. The general architecture of the end-to-end system is depicted in Fig. 1.

The end-to-end system is trained via one-class softmax objective, which can be formulated as [11]:

$$L_{OCS} = -\frac{1}{N}\sum_{i=1}^{N} log(1 + e^{k(m_{y_i} - \hat{W}_0\hat{\omega}_i)(-1)^{y_i}}) \quad (1)$$

where k is the scale factor, $\omega_i \in R^D$ and $y_i \in \{0,1\}$ are the D-dimensional embedding vector and label of the i^{th} sample respectively. N is the mini-batch size and m_{y_i} defines the compactness margin for class label y_i. The larger is the margin, the more compact the embeddings will be. W_0 is the weight vector of our target class embeddings. Both $\hat{W}_0$ and $\hat{\omega}_i$ are normalizations of W_0 and ω_i respectively.

3. Conventional activation functions

3.1. Non-trainable activation functions

For the past several years, the ReLU activation function was widely used in various neural network systems in speaker verification and end-to-end spoof detection. The ReLU function is defined as follows:

$$f_{ReLU}(x_i) = \begin{cases} x_i & \text{if } x_i > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where x_i is the input feature. The main reason behind the populatiry of the ReLU activation is that ReLU can overcome the vanishing gradient problem and enables the model to train faster. Although the introduction of the ReLU activation have shown impressive performance in various tasks, due to its inability to handle negative inputs, it was occasionally reported that the ReLU can limit the system's ability to learn.

In order to solve the limitations of ReLU, various variants were proposed, such as the LeakyReLU, exponential LU (ELU) and randomized ReLU (RReLU). The formulation for LeakyReLU, ELU and RReLU are as follows:

$$f_{LeakyReLU}(x_i) = \begin{cases} x_i & \text{if } x_i > 0 \\ \gamma x_i & \text{otherwise,} \end{cases} \quad (3)$$

$$f_{ELU}(x_i) = \begin{cases} x_i & \text{if } x_i > 0 \\ r(e^{x_i} - 1) & \text{otherwise,} \end{cases} \quad (4)$$

where γ and r are fixed parameters. The RReLU is defined similarly to the LeakyReLU as follows:

$$f_{RReLU}(x_i) = \begin{cases} x_i & \text{if } x_i > 0 \\ a_i x_i & \text{otherwise,} \end{cases} \quad (5)$$

where a_i is randomly sampled from $unif(l,u)$ while training, and l and u are fixed parameters. When testing, the a_i is set to the mean of $unif(l,u)$, which is $\frac{l+u}{2}$. As described in equation 3, 4, 5, the ReLU variants commonly focus on taking the negative inputs into consideration by giving a slope to the negative region.

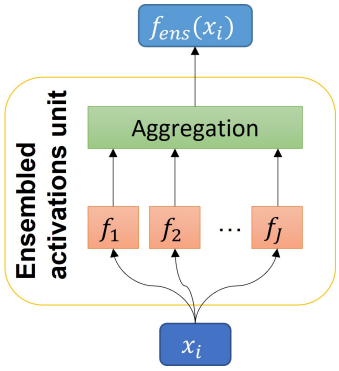

Figure 2: The general diagram for the activation ensemble framework.

3.2. Trainable activation functions

In order to allow the activation functions to operate in a data-adaptive manner, several learnable activation functions were proposed. One of them is the parametric ReLU (PReLU), which operates in a similar fashion to the LeakyReLU, but uses a trainable parameter for the negative slope:

$$f_{PReLU}(x_i) = \begin{cases} x_i & \text{if } x_i > 0 \\ \xi_i x_i & \text{otherwise,} \end{cases} \quad (6)$$

where ξ_i is a learnable parameter.

The attention ReLU (AReLU) goes one step further than the other ReLU variants, by employing a trainable attention mechanism to boost the contribution of relevant input features while suppressing the irrelevant ones [19]. More specifically, the AReLU is a combination of the standard ReLU and the element-wise sign-based attention (ELSA). Given input x_i, which is the i^{th} element of feature X, the AReLU is fomulated as follows:

$$f_{AReLU}(x_i) = f_{ReLU}(x_i) + g_{att}(x_i, \alpha, \beta) \quad (7)$$

$$= \begin{cases} C(\alpha)x_i & \text{if } x_i < 0 \\ (1 + \sigma(\beta))x_i & \text{otherwise,} \end{cases} \quad (8)$$

where α and β are learnable scaling parameters, C is the clamping operation which restricts the value to $[0.01, 0.99]$, and σ is the sigmoid function. While the β parameter amplifies the positive elements, the α parameter suppresses the negative elements.

4. Activation ensemble framework

As described in equations 2–8, each activation operates and focuses on different aspects of the feature map. Therefore, in order to exploit the complementary information propagated through the different activations, we propose to apply activation ensemble [21] to the end-to-end system. Instead of using a single type of activation function, the activation ensemble scheme uses multiple activation functions and pools their outputs via summation:

$$f_{ens}(x_i) = \sum_{j=1}^{J} f_j(x_i), \quad (9)$$

where f_j is an activation and J is the number of unique activation functions used. The general framework of the activation ensemble framework is depicted in Figure 2.

Table 2: Summary of ASVspoof2019 logical Access (LA) corpora in terms of training (Train), development (Dev) and evaluation (Eval) partitions and number of recordings.

	#Speakers	#Recordings	
		Bona fide	Spoof
Training partition	20	2,580	22,800
Development partition	20	2,548	22,296
Evaluation partition	67	7,355	63,882

Table 3: Initial values for the parameters of the activation functions.

ξ_i (PReLU)	γ (LeakyReLU)	r (ELU)	l (RReLU)	u (RReLU)
0.25	0.2	1.0	0.125	0.333

5. Experiments

5.1. Experimental setup

As local frame-level frame hand-crafted features, we use 60-dimensional linear frequency cepstral coefficients (LFCC) extracted using 25ms analysis window over a frame shift of 10ms. No data augmentation was performed in our experiments.

For training and evaluating the experimented systems, the ASVspoof 2019 challenge dataset was used, which provides a common framework with a standard corpora for conducting spoofing detection research on LA attacks. The LA dataset includes bonafide and spoof speech signals generated using various state-of-the-art voice conversion and speech synthesis algorithms. A summary of the LA corpora in terms of training (Train), development (Dev) and evaluation (Eval) partitions and number of recordings is presented in Table 2. The development and evaluation subsets constitute the seen and unseen test sets in terms of spoofing attacks. For more details about the corpora, the interested readers are referred to [22]. For training all the experimented systems, balanced mini-batches of size 64 samples were used. The ADAM optimizer was used with initial learning rate of 0.0003 and exponential learning rate decay with rate of 0.5 was applied [11, 18].

For comparing the performance of different systems, the official evaluation metrics of ASVspoof2019 challenge, equal error rate (EER) and minimum tandem detection cost function (min-tDCF) [23], were used. The lower the values of EER and min-tDCF, the better performance is attained. The ASV scores provided by the challenge organizer were used for computing min-tDCF.

5.2. Experimental results

5.2.1. Comparison between different ReLU variants

In this experiment, we compare the performance of end-to-end systems with different activation functions. More specifically, we compare 6 types of ReLU-based activation functions: ReLU, LeakyReLU [24], Randomized LeakyReLU (RReLU) [25], exponential linear unit (ELU) [26], parametric ReLU (PReLU) [27], and AReLU. The initial values for the parameters of each activation function is described in Table 3.

Table 4 shows the EER and min-tDCF results of the experimented systems with different ReLU-based activation. As depicted in the results, all activation functions were able to perform well (i.e., EER less than 1%) on the development set, which includes known attacks only. While the standard ReLU

Table 4: The experimental results of the SE-ResNet-18-based end-to-end systems with different activation functions on the ASVSpoof2019 Logical Access Development and Evaluation sets.

	Dev	Eval	
	EER [%]	EER [%]	min-tDCF
ReLU	0.1082	3.0589	0.0718
LeakyReLU	0.2388	2.7999	0.0696
RReLU	0.2697	3.2104	0.0790
ELU	0.2366	4.7026	0.0980
PReLU	0.1480	2.6515	0.0663
AReLU	0.2433	**2.3770**	**0.0586**

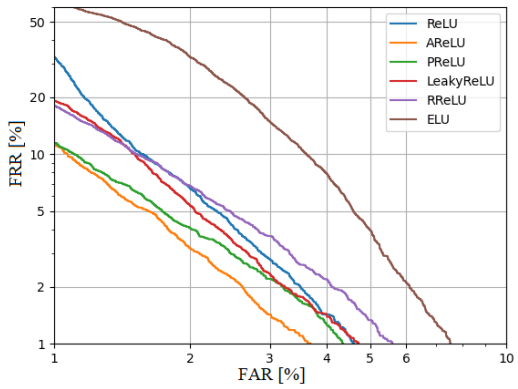

Figure 3: The DET curves of the SE-ResNet-18-based systems with different activation functions on the ASVSpoof2019 Logical Access Evaluation set.

activation achieved best result on the development set, it could be seen that 3 variants of the ReLU (i.e., LeakyReLU, PReLU, AReLU) were able to outperform the standard ReLU activation on the evaluation set. From this result, we could assume that allowing the negative elements can help the network to generalize better on unknown attacks by focusing more on the relevant features for the spoof detection task. However, as seen in the ELU and RReLU results, blindly passing the negative elements with no knowledge on the dataset does not always guarantee good performance.

Another interesting point to notice from the results is that the learnable activation functions (i.e., PReLU, AReLU) were able to perform better than the fixed activations (i.e., ReLU, LeakyReLU, RReLU, ELU). This may be attributed to the fact that the learnable activation functions are more capable of suppressing the nuisance features as their scaling parameters are optimized in a data-adaptive fashion.

Among the learnable activations (i.e., PReLU, AReLU), the AReLU achieved the best performance, which outperformed the PReLU with a relative improvement of 11.61% in terms of min-tDCF. Although the PReLU suppresses the negative elements in a similar manner to the AReLU, it does not attempt to amplify the relevant features. Therefore the AReLU may be more suited to focus on the artifacts caused by the generation process of the spoof attacks (e.g., speech synthesis, voice conversion), as it can emphasize the positive elements via learnable scaling parameter β.

The DET curves of the experimented systems are depicted in Fig. 3.

Table 5: The experimental result of the single activation and the activation ensembled systems on the ASVSpoof2019 Logical Access Development and Evaluation sets.

	Dev	Eval	
	EER [%]	EER [%]	min-tDCF
ReLU+AReLU	0.1968	2.3655	**0.0519**
ReLU+AReLU+PReLU	0.1968	2.5700	0.0658
AReLU+PReLU	0.2366	2.3625	0.0565
AReLU+LeakyReLU	0.1968	2.3410	0.0630
AReLU+ELU	0.1968	2.2464	**0.0550**
ReLU+AReLU+PReLU+LeakyReLU+ELU	0.3538	**2.2148**	0.0575

5.2.2. Activation ensemble for end-to-end spoofing countermeasure system

Analogous to the single activation systems, the ensemble system was trained in an end-to-end fashion, taking the LFCC features as input. Table 5 shows the performance of the activation ensembled systems with different combinations of activation functions. As depicted in the results, although the performance on the development set was slightly degraded compared to the standard ReLU activation, the ReLU-based system could benefit greatly in terms of detecting unseen attacks by using learnable activation functions (e.g., AReLU, PReLU) in conjunction via activation ensemble. Especially using the ReLU and AReLU together achieved a relative improvement of 22.67% in terms of EER over the ReLU-based system on the evaluation set. Similarly, the performance of the AReLU-based system could be improved by ensembling various non-learnable activation functions (e.g., LeakyReLU, ELU). Especially when using the AReLU and ELU achieved a relative improvement of 6.14% in terms of min-tDCF over the AReLU-based system. The best performance in terms of EER was observed by ensembling the ReLU, AReLU, PReLU, LeakyReLU and ELU.

6. Conclusion

In this paper, we investigate the effects of different activation functions employed in an end-to-end spoof detection system. More specifically, we experimented with several variants of the conventional rectified linear unit (ReLU) activation function on the ASVSpoof2019 Challenge logical access dataset, and analyzed their performance to capture the artifacts created by the spoof generation process.

Our results showed that using learnable activation functions, such as parametric ReLU (PReLU) or attention ReLU (AReLU) can greatly improve the anti-spoofing countermeasure performance over the non-learnable activation functions (i.e., ReLU, LeakyReLU, randomized ReLU, exponential LU). Moreover, in order to fully consider the complementary information learned by each activation, we have proposed an end-to-end system with multiple activation functions via activation ensemble. From our results, we could see that ensembling multiple different activation functions, including the learnable and non-learnable ones, can greatly improve the performance in terms of spoof detection.

In our future study, we will be expanding the AReLU activation function to be more suited for finding the artifacts within the given speech spectrum. Moreover, we will be exploring a more effective way to exploit the complementary information learned via different activation functions. Furthermore, we will be evaluating the end-to-end systems with attentive activation function on other spoofing attack types, such as the physical

access spoofing attack.

7. Acknowledgment

We wish to acknowledge funding from the Natural Sciences and Engineering Research Council of Canada (NSERC) through grant RGPIN-2019-05381 and Ministry of Economy and Innovation (MEI) of the Government of Quebec for the continued support.

8. References

[1] M. Todisco, H. Delgado, and N. Evans, "Constant q cepstral coefficients: A spoofing countermeasure for automatic speaker verification," *Computer Speech & Language*, vol. 45, pp. 516–535, 2017.

[2] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," in *Proc. Interspeech 2015*, 2015, pp. 2087–2091.

[3] J. Alam and P. Kenny, "Spoofing detection employing infinite impulse response—constant q transform-based feature representations," in *Proc. 2017 25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 101–105.

[4] X. Tian, Z. Wu, X. Xiao, E. S. Chng, and H. Li, "Spoofing detection from a feature representation perspective," in *Proc. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2119–2123.

[5] T. Patel and H. Patil, "Combining evidences from mel cepstral, cochlear filter cepstral and instantaneous frequency features for detection of natural vs. spoofed speech," in *Proc. Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[6] T. Patel and H. Patil, "Effectiveness of fundamental frequency (f 0) and strength of excitation (soe) for spoofed speech detection," in *Proc. 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5105–5109.

[7] X. Xiao, X. Tian, S. Du, H. Xu, E. S. Chng, and H. Li, "Spoofing speech detection using high dimensional magnitude and phase features: The ntu approach for asvspoof 2015 challenge," in *Proc. Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[8] J. Alam, P. Kenny, V. Gupta, and T. Stafylakis, "Spoofing detection on the asvspoof2015 challenge corpus employing deep neural networks," in *Proc. Odyssey 2016 The Speaker and Language Recognition Workshop*, 2016, pp. 270–276.

[9] J. Monteiro and J. Alam, "Development of voice spoofing detection systems for 2019 edition of automatic speaker verification and countermeasures challenge," in *Proc. 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 1003–1010.

[10] H. Tak, J. Patino, A. Nautsch, N. Evans, and M. Todisco, "Spoofing Attack Detection Using the Non-Linear Fusion of Sub-Band Classifiers," in *Proc. Interspeech 2020*, 2020, pp. 1106–1110.

[11] Y. Zhang, F. Jiang, and Z. Duan, "One-class learning towards synthetic voice spoofing detection," *IEEE Signal Processing Letters*, vol. 28, pp. 937–941, 2021.

[12] P. RahulT, P. R. Aravind, C. Ranjith, U. Nechiyil, and N. Paramparambath, "Audio spoofing verification using deep convolutional neural networks by transfer learning," *ArXiv*, vol. abs/2008.03464, 2020.

[13] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "STC Antispoofing Systems for the ASVspoof2019 Challenge," in *Proc. Interspeech 2019*, 2019, pp. 1033–1037.

[14] Z. Wu, R. K. Das, J. Yang, and H. Li, "Light Convolutional Neural Network with Feature Genuinization for Detection of Synthetic Speech Attacks," in *Proc. Interspeech 2020*, 2020, pp. 1101–1105.

[15] T. Chen, A. Kumar, P. Nagarsheth, G. Sivaraman, and E. Khoury, "Generalization of Audio Deepfake Detection," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 132–137.

[16] J. Monteiro, J. Alam, and T. Falk, "A multi-condition training strategy for countermeasures against spoofing attacks to speaker recognizers," in *Proc. Odyssey 2020 The Speaker and Language Recognition Workshop*, 2020, pp. 296–303.

[17] H. Tak, J. Patino, M. Todisco, A. Nautsch, N. Evans, and A. Larcher, "End-to-end anti-spoofing with rawnet2," in *Proc. 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, Ed., 2021.

[18] J. Monteiro, J. Alam, and T. H. Falk, "Generalized end-to-end detection of spoofing attacks to automatic speaker recognizers," *Computer Speech & Language*, p. 101096, 2020.

[19] D. Chen and K. Xu, "Arelu: Attention-based rectified linear unit," *arXiv preprint arXiv:2006.13858*, 2020.

[20] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.

[21] D. Klabjan and M. Harmon, "Activation ensembles for deep neural networks," in *Proc. 2019 IEEE International Conference on Big Data*, 2019, pp. 206–214.

[22] J. Yamagishi, M. Todisco, M. Sahidullah, H. Delgado, X. Wang, N. Evans, T. Kinnunen, K. A. Lee, V. Vestman, and A. Nautsch, "Asvspoof 2019: The 3rd automatic speaker verification spoofing and countermeasures challenge database," 2019.

[23] 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, "t-dcf: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification," *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 2195–2210, 2020.

[24] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. International Conference on Machine Learning*, 2013.

[25] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arxiv:1505.00853*, 2015.

[26] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units

(elus)," in *Proc. 4th International Conference on Learning Representations, ICLR 2016*, 2016.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," *arXiv preprint arxiv:1502.01852*, 2015.