Defense for Black-box Attacks on Anti-spoofing Models by Self-Supervised Learning

Haibin Wu[1], *Andy T. Liu*[12], *Hung-yi Lee*[12]

[1]Graduate Institute of Communication Engineering, National Taiwan University
[2]College of Electrical Engineering and Computer Science, National Taiwan University
{f07921092, r07942089, hungyilee}@ntu.edu.tw

Abstract

High-performance anti-spoofing models for automatic speaker verification (ASV), have been widely used to protect ASV by identifying and filtering spoofing audio that is deliberately generated by text-to-speech, voice conversion, audio replay, etc. However, it has been shown that high-performance anti-spoofing models are vulnerable to adversarial attacks. Adversarial attacks, that are indistinguishable from original data but result in the incorrect predictions, are dangerous for anti-spoofing models and not in dispute we should detect them at any cost. To explore this issue, we proposed to employ Mockingjay, a self-supervised learning based model, to protect anti-spoofing models against adversarial attacks in the black-box scenario. Self-supervised learning models are effective in improving downstream task performance like phone classification or ASR. However, their effect in defense for adversarial attacks has not been explored yet. In this work, we explore the robustness of self-supervised learned high-level representations by using them in the defense against adversarial attacks. A layerwise noise to signal ratio (LNSR) is proposed to quantize and measure the effectiveness of deep models in countering adversarial noise. Experimental results on the ASVspoof 2019 dataset demonstrate that high-level representations extracted by Mockingjay can prevent the transferability of adversarial examples, and successfully counter black-box attacks.

Index Terms: adversarial attack, black-box attack, anti-spoofing, ASV, self-supervised learning

1. Introduction

Automatic speaker verification, abbreviated as ASV, is the task to verify whether a piece of speech sample belongs to a certain speaker. Thanks to the efforts of previous researchers [1–5], it is now a matured technology widely applied to biometric identification. However, evidence shows that unprotected ASV models are highly vulnerable to spoofing audio deliberately generated by text-to-speech, voice conversion, and audio replay [6, 7], as some malicious attackers mimic a specific target user to deceive the ASV systems. As a result, strategies to handle the spoofing audio are in need. The ASVspoof challenge series [8–10] is a community-driven challenge to arouse attention in addressing spoofing audio attacks and their countermeasures. Spoofing countermeasure models, also known as anti-spoofing models, are shields for ASV to detect and filter spoofing audio. Recently several high performance anti-spoofing models have been proposed [11–20].

Since Szegedy et al. [21] first proposed the concept of adversarial attacks, and illustrate how deep neural networks with impressive performance for computer vision tasks are vulnerable to adversarial attacks, a large variety of research in this domain have been done. Adversarial example, which is generated by adding imperceptible perturbation to the input sample, can result in the incorrect prediction of the models. The added perturbation is carefully crafted such that humans can not distinguish the adversarial example from the input sample visually or acoustically. Attacking the models with adversarial examples is called adversarial attack. Previous works show that deep neural networks for speech processing tasks are subject to adversarial attacks. [22] investigates the vulnerability of automatic speech recognition (ASR) models to adversarial attacks. Given any audio waveform, whether speech or music, they can generate an adversarial example, which is over 99% similar to the original audio but makes the ASR model wrongly predict any transcribes they defined before. It has also been shown that ASV systems can be fooled by adversarial examples [23, 24]. Moreover, in [25, 26], the authors illustrate the anti-spoofing models for ASV systems are also vulnerable to adversarial attacks. In this paper, we focus on the defense for adversarial attacks of the anti-spoofing models.

Proactive defense and passive defense are two main categories of defense for the adversarial attacks. The former defense aims to train new models to counter the adversarial attacks. The most famous proactive defense method is adversarial training [27], which injects the adversarial examples generated by different attack algorithms into the training data. It is reasonable that the models are robust to specific attack algorithms if the models have already seen the adversarial examples during training. However, adversarial training is time-consuming and resource-consuming. Whats more, when defenders do adversarial training, they have no idea which attack algorithm the attackers will take. The mismatch between attack algorithms during training and inference will make the models susceptible to adversarial attacks they havent seen during training. Passive defense methods embrace the advantage of defending adversarial examples generated by all kinds of attacking methods without modifying the model. The spatial smoothing, also called as filter, is a passive defense method that counters adversarial attacks [28]. Gaussian filter, median filter and mean filter are used to defend anti-spoofing models in [25]. The above three filters are shallow filters to counter the deliberately generated adversarial perturbations.

In this paper, we propose a passive defense leveraging the power of self-supervised learning. Self-supervised learning allows the model to learn high-level and contextualized representations from a large amount of unlabeled data, without the use of any label [29]. In self-supervised learning, a pre-training task (or auxiliary task) that uses only unlabeled data is formulated, and the model is required to solve such a task. While solving the pre-training task, the model is also learning a function that maps input to high-level representations, which can

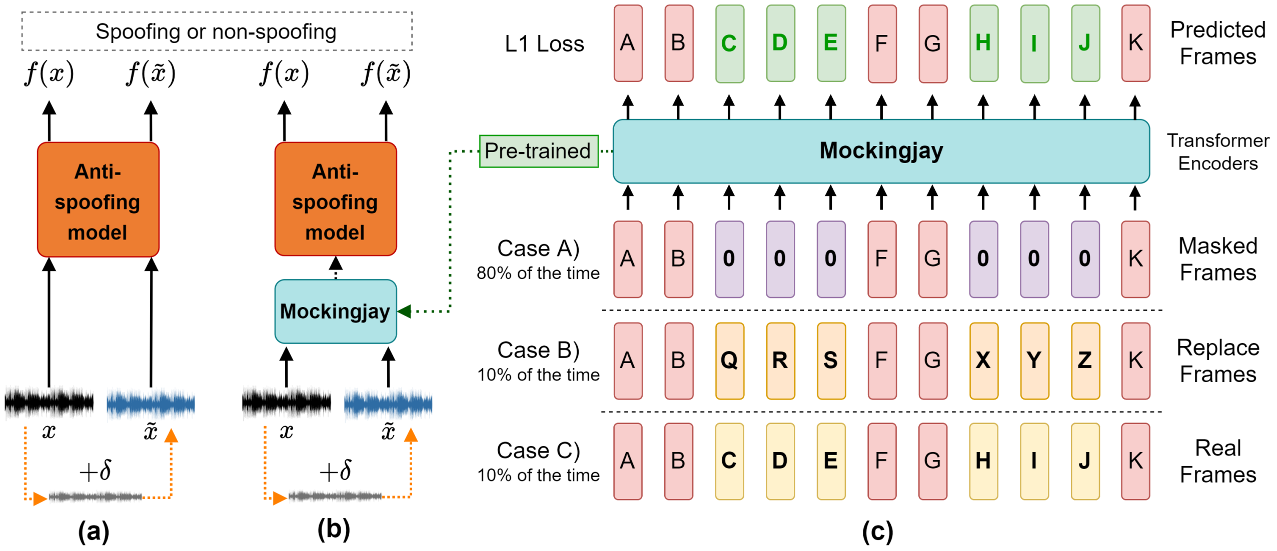

Figure 1: *(a) Adversarial attack, (b) Proposed method, (c) Illustration of the self-supervised Mockingjay pre-training.*

potentially transfer information learned from unlabeled data to downstream tasks. Through pre-training models on speech, self-supervised learning based models are able to leverage the knowledge of unlabeled speech, then the performance of downstream speech and language processing (SLP) tasks can be improved dramatically [30–33], including phone classification, speaker recognition, and speech recognition. However, the robustness of such high-level audio representations learned by self-supervised learning based models against adversarial attacks for anti-spoofing of ASV have not been studied yet.

In this work, we find that self-supervised learning models can serve as a deep filter which extracts the pivotal information from the contaminated input spectrograms to counter the adversarial attacks. To the best of our knowledge, we are among the first ones to adopt the high-level representations extracted by the self-supervised model for the defense of anti-spoofing models in the black-box scenario and the experimental results show the effectiveness of our proposed method. We also propose the layerwise noise to signal ratio (LNSR), to quantize and measure the effect of deep models in countering adversarial noise. We find that the adversarial noise is attenuated layer by layer in the self-supervised learning model.

2. Adversarial attack

When a tiny perturbation, which is imperceptible to humans, is deliberately crafted and added to the original example, the new example will lead to the model's incorrect prediction. We call the new example and the tiny perturbation as adversarial example and adversarial noise respectively.

As shown in Figure 1 a), x is the original example, δ is the adversarial noise and $\tilde{x}$ is the adversarial example. Given the anti-spoofing model $f(.)$, we denote the prediction of the original example and adversarial example as $f(x)$ and $f(\tilde{x})$. The adversarial example is generated as this equation:

$$\tilde{x} = x + \delta. \tag{1}$$

Finding an adversarial example is equivalent to crafting a suitable perturbation δ and searching for a suitable δ is an optimization problem as shown below:

$$max_{\|\delta\|_\infty \leq \epsilon} Diff(f(x), f(\tilde{x})), \tag{2}$$

where $Diff(f(x), f(\tilde{x}))$ means the difference between $f(x)$ and $f(\tilde{x})$ and it is totally differentiable, $\|\delta\|_\infty$ is the L_∞ norm, ϵ is a constant we defined to constrain δ. We solve the optimization problem by doing gradient descent to the input with the model parameters fixed. We want the adversarial example x to be as similar as $\tilde{x}$ to make them indistinguishable by human, so the noise δ shouldn't be too large. In this paper, δ is constrained in an L_∞ norm ball. Different searching strategies for δ result in different attack algorithms. In this paper, we adopted the fast gradient-sign method (FGSM) [27] and the projected gradient descent method (PGD) [34].

There are two kinds of adversarial attack scenarios: black-box attack and white-box attack. In both two scenarios, there are two models: the *target* model and the *attacking* model. The target model is the model attackers aim to attack, and the attacking model is the model implemented by the attackers to generate adversarial examples. In the black-box attack scenario, the target model and the attacking model are different models, while in the white-box attack scenario, the target model is also the attacking model. In the white-box attack scenario, the attackers know everything about the target model, including model parameters, gradients, etc. Obviously it is unrealistic that the attackers have all access to the target model. In the black-box scenario, the attackers can't obtain the inner parameters of the target model, while they can collect the inputs and the outputs of the target model by querying it. Then the attackers will train a substitute model and employ the substitute as the attacking model to generate adversarial samples with transferability. Our objective in this paper is to prevent the transferability of such adversarial examples, and improve the robustness of anti-spoofing models against black-box attacks by leveraging high-level representations extracted by self-supervised models.

3. Proposed method

3.1. Mockingjay

The Mockingjay [30] approach learns representations of speech by solving a self-supervised masked-prediction task with a L_1

reconstruction loss function. The model is based on multi-layer transformer encoders with multi-head self-attention [35] followed by a feed forward prediction network. The transformer encoder produces a representation vector for each time frame, and the prediction network reconstructs frames of spectrogram to solve the masked-prediction task. At training time, the masked-prediction task requires the model to take a sequence of frames as input that has had a certain percentage of randomly selected frames masked, and attempts to reconstruct the masked frames. After training, the representations produced by the transformer network are inputs to the anti-spoofing model instead of acoustic features. We illustrate this in Figure 1 b).

We consider a masking policy following [29, 30]. The following three cases are sampled. Case A) we mask all selected frames to zero; this happens for 80% of the time. Case B) we replace all selected frames with random frames, 10% of the time. And Case C) we leave all the selected frames untouched for the rest 10% of the time. Then the sampled case (A, B, or C) would be applied on 15% of randomly selected frames. The intuition is that by reconstructing from corrupted input, the model should learn a solid understanding of the high-level content, which provides immunity to adversarial attacks. To adapt the local smoothness of acoustic sequence, we mask contiguous segments of C_{num} of frames. We illustrate this in Figure 1 c), where we show an example of $C_{num} = 3$. We also employ the downsampling technique from [30], where we reshape and stack R_{factor} consecutive frames into one step.

3.2. Self-Supervised Learned Adversarial Defender

In this work, we propose to adopt the Mockingjay to protect the anti-spoofing models. Superficial or surface features like Mel-Spectrogram often buries the abundant information of speech, extracting representations with the Mockingjay transform thus makes the high-level information more accessible to downstream tasks. We first extract the high-level representations from spectrograms by Mockingjay and then use the high-level features to train the anti-spoofing model. The cascade of the Mockingjay and anti-spoofing model shown in Figure 1 b) is called self-supervised learned adversarial defender.

In the black-box attack scenario, the attackers are not aware of the existence of the Mockingjay and only know the inputs to the target system are spectrograms. They attempt to nd adversarial noise to add it to the input spectrograms by using the attacking model. However, before the input spectrograms are thrown into the anti-spoofing model, the Mockingjay will help alleviate the superficial noise added to the input spectrograms and avoid the transferability of adversarial noise. Experimental results show that the high-level representation extracted by Mockingjay prevents the transferability of adversarial noise and counter the black-box attacks.

The readers may challenge the power of defense here comes from the mismatch of the network architecture between the target and attacking model. The target model has the Mockingjay as the front-end, while the attacking model does not. It may be intuitive that the attack signal for the attacking model cannot transfer to the target model. However, experimental results show that pre-training plays a critical role in the defense. Without the pre-training, merely using the mismatch of network architecture can not avoid adversarial noise's transferability.

There are two plausible reasons that the Mockingjay can help counter the adversarial noise. From the perspective of the self-supervised training procedure of the Mockingjay, the masked-prediction task introduces noise to the input spectrograms. The Mockingjay is trained to learn how to weaken the noise in the input spectrograms, extract pivotal information from the contaminated spectrograms, and use the pivotal information to reconstruct the original clean spectrograms. The adversarial noise is also a kind of noise to some extent. So in our proposed approach, the Mockingjay would weaken the adversarial noise added to the input spectrograms, extract key information and pass the key information to the anti-spoofing model to finish the anti-spoofing task. From the loss function perspective, in the black-box attack scenario, usually the target model and the attacking model perform the same task and are trained by classification loss. It is intuitively the adversarial perturbations generated by the attacking model are with transferability to the target model as they are both sensitive to classification loss. However, in the proposed approach, the Mockingjay is trained by reconstruction loss and performs the task which is different from the attacking model.

3.3. Layerwise noise to signal ratio

We propose a measurement named layerwise noise to signal ratio ($LNSR$) in order to estimate the intensity of adversarial noise in different layers of the Mockingjay:

$$LNSR_i = \sum_{n=1}^{N} \frac{\|\hat{h}_i^n - h_i^n\|_2}{\|h_i^n\|_2} \quad for\, i = 0, 1, \ldots, K, \tag{3}$$

where K is the total layer number of the Mockingjay, N is the number of the adversarial example - original example pairs, $\|.\|_2$ means L_2 norm, $\hat{h}_i^n$ and h_i^n are the features of the i^{th} layer of the adversarial example and original example respectively for the n^{th} pair. When $i = 0$, $\hat{h}_i^n$ and h_i^n are the adversarial example and original example themselves. $LNSR_i$ aims to measure the amount of the attack signals in each layer. If the value of $LNSR_i$ decreases when i increases, that means Mockingjay attenuates the attacking noises.

4. Experiment

4.1. Experiment setup

For the dataset, we use the LA partition of the ASVspoof 2019 challenge, which contains fake audios generated by text to speech and voice conversion. The dataset is itself divided into three parts: training, development, and evaluation.

For the Mockingjay, we use the prevailing framework of the LARGE model described in Mockingjay [30], which consists of 12 layers of Transformer Encoders. We follow the pre-training settings as in [30], where we pre-train our Mockingjay model on 360 hours of speech on the LibriSpeech dataset [36]. Two high-performance anti-spoofing models are adopted: LCNN [19] and SENet [20]. The implementation details of the two models can be found in [26]. We refer to the LCNN and SENet trained by the mel-spectrograms as basic LCNN and SENet. In the black-box attack scenario, we use the basic LCNN and SENet as the attacking models to generate adversarial examples and use the models in Figure 2 as target models. The adversarial examples generated by the basic LCNN are used to attack the basic SENet and its variants. The adversarial examples generated by the basic SENet are used to attack the basic LCNN and its variants. We use FGSM and PGD as attack algorithms, and we measure over different values of ϵ: $0.1, 1, 2, 4, 8, 16$.

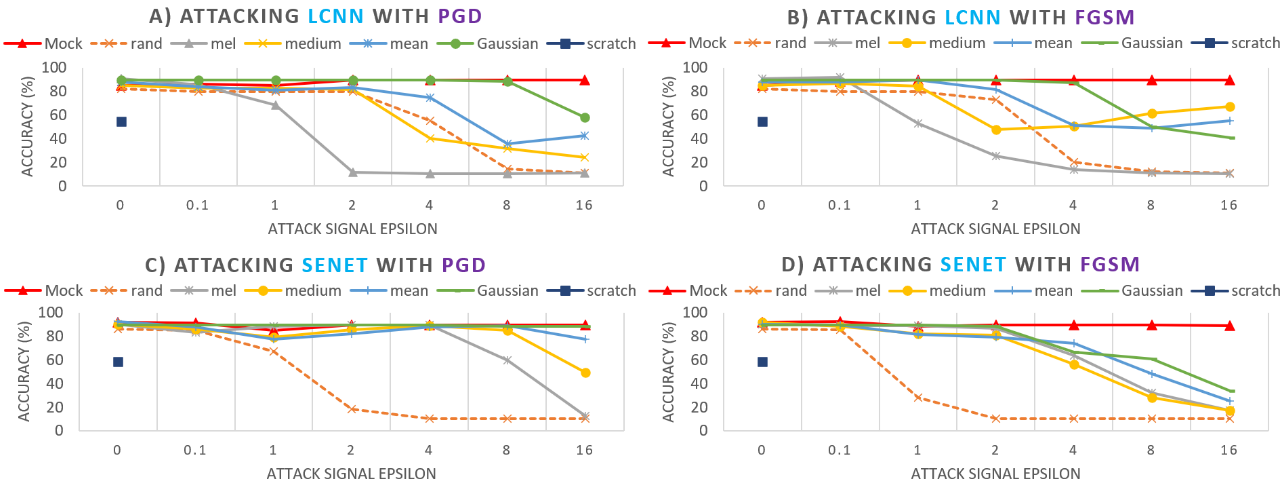

Figure 2: *Comparison of different defense methods against two attack algorithms over increasing amount of attack signal* ϵ.

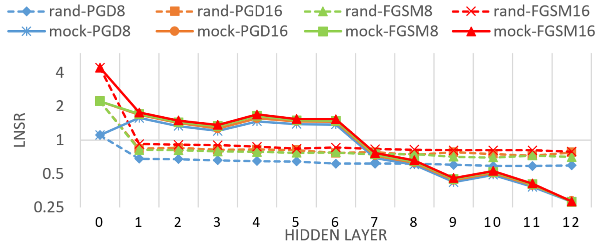

Figure 3: *Layerwise noise to signal ratio on Mockingjay*

4.2. Result and analysis

4.2.1. *Comparing different defense approaches*

The proposed approach is compared with various passive filter-based defense approaches [25]. Results are presented in Figure 2. We refer to the cascade of the pre-trained Mockingjay and anti-spoofing model as Mock, the basic LCNN or SENet as Mel, and finally the anti-spoofing models equipped with different hand-designed filters as medium, mean, and Gaussian. In Figure 2 A) and B), LCNN and its variants are attacked by PGD and FGSM, respectively. In Figure 2 C) and D), SENet and its variants are attacked by PGD and FGSM, respectively.

As expected, basic Mel models (grey curve) are vulnerable to adversarial attacks. In all four scenarios, we see the proposed Mockingjay defense mechanism prevails over all other approaches (red curve, denoted as Mock), as Mockingjay is invariant to adversarial attacks. The attack always fails no matter the amount of attack signal. Although Mockingjay is pre-trained on LibriSpeech [36] but not ASVspoof data, it is still capable of leveraging self-supervised learned knowledge to defense adversarial attacks. Other filters (medium, mean, and Gaussian) also counter the attack to some extent, but cannot resist high values of ϵ as Mockingjay. The Mockingjay model outperforms all the filters in all circumstances.

Moreover, we show results of a random parameterized Mockingjay (orange curve, denoted as rand) to demonstrate the effect of pre-training. Random parameterized Mockingjay also shows some capability of defense in Figure 2 A) and B), but fails to protect anti-spoofing models against adversarial attacks as ϵ increases. It completely fails to protect anti-spoofing models in Figure 2 C) and D). The results show that the random Mockingjay is much worse than the pre-trained model, and even worse than the hand-designed filters in some cases. This shows that our success in the red curve (Mock) is not simply from the mismatch of network architecture between the target model and attacking model. To further show the importance of pre-training, we trained the anti-spoofing models with the same architecture as the cascade of Mockingjay and LCNN/SENet from scratch. The results are denoted as “scratch” (dark blue dot). The results show that training the cascade model from scratch results in a low accuracy that barely surpasses random guesses. This further shows that the success in the red curve (Mock) is not contributed by model size.

4.2.2. *Measuring the removal of adversarial noise*

The values of $LNSR$ (Section 3.3) in different layers of different models are shown in Figure 3. rand-PGDϵ and mock-PGDϵ means we use the adversarial examples generated by PGD with ϵ to calculate the $LNSR$ on the random parameterized Mockingjay and pre-trained Mockingjay, respectively. Two ϵ values are tested: $8, 16$. From Figure 3, the pre-trained Mockingjay successfully lowers the $LNSR$. When the model becomes deeper, the value of $LNSR$ becomes lower, which illustrates the effect of the Mockingjay to alleviate the adversarial noise. In contrast, the random parameterized Mockingjay can only reduce $LNSR$ to a certain degree. When model depth increases, the value of $LNSR$ is quickly saturated. This is another evidence to show the importance of pre-training in defense.

5. Conclusion

In this work, we propose to use a self-supervised learning model to protect the anti-spoofing models against black-box attacks. Experimental results illustrate the representations extracted by self-supervised learning model prevent the transferability of adversarial examples and counter the black-box attacks. The proposed layerwise noise to signal ratio manifests the effectiveness of the self-supervised learning model in alleviating the adversarial noise layer by layer. For the future work, we would like to explore the capability of defense for the self-supervised model learned with different objectives and apply the proposed defense approaches on more speech processing applications.

6. References

[1] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[2] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Twelfth annual conference of the international speech communication association*, 2011.

[3] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[4] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2014, pp. 1695–1699.

[5] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, "Deep neural networks for extracting baum-welch statistics for speaker recognition," in *Proc. Odyssey*, 2014, pp. 293–298.

[6] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *speech communication*, vol. 66, pp. 130–153, 2015.

[7] Z. Wu, P. L. De Leon, C. Demiroglu, A. Khodabakhsh, S. King, Z.-H. Ling, D. Saito, B. Stewart, T. Toda, M. Wester *et al.*, "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 768–783, 2016.

[8] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, "Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[9] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, and K. A. Lee, "The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," 2017.

[10] 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," *arXiv preprint arXiv:1904.05441*, 2019.

[11] N. W. Evans, T. Kinnunen, and J. Yamagishi, "Spoofing and countermeasures for automatic speaker verification." in *Interspeech*, 2013, pp. 925–929.

[12] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant q cepstral coefficients," *Odyssey 2016*, Jun 2016. [Online]. Available: http://dx.doi.org/10.21437/Odyssey.2016-41

[13] H. Delgado, M. Todisco, M. Sahidullah, N. Evans, T. Kinnunen, K. Lee, and J. Yamagishi, "Asvspoof 2017 version 2.0: metadata analysis and baseline enhancements," in *Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018.

[14] Z. Wu, T. Kinnunen, E. S. Chng, H. Li, and E. Ambikairajah, "A study on spoofing attack in state-of-the-art speaker verification: the telephone speech case," in *Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference*. IEEE, 2012, pp. 1–5.

[15] N. Chen, Y. Qian, H. Dinkel, B. Chen, and K. Yu, "Robust deep feature for spoofing detectionthe sjtu system for asvspoof 2015 challenge," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[16] Z. Chen, Z. Xie, W. Zhang, and X. Xu, "Resnet and model fusion for automatic spoofing detection." in *INTERSPEECH*, 2017, pp. 102–106.

[17] Y. Qian, N. Chen, and K. Yu, "Deep features for automatic spoofing detection," *Speech Communication*, vol. 85, pp. 43–52, 2016.

[18] C.-I. Lai, A. Abad, K. Richmond, J. Yamagishi, N. Dehak, and S. King, "Attentive filtering networks for audio replay attack detection," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6316–6320.

[19] G. Lavrentyeva, S. Novoselov, A. Tseren, M. Volkova, A. Gorlanov, and A. Kozlov, "Stc antispoofing systems for the asvspoof2019 challenge," *arXiv preprint arXiv:1904.05576*, 2019.

[20] C.-I. Lai, N. Chen, J. Villalba, and N. Dehak, "Assert: Antispoofing with squeeze-excitation and residual networks," *arXiv preprint arXiv:1904.01120*, 2019.

[21] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[22] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2018, pp. 1–7.

[23] F. Kreuk, Y. Adi, M. Cisse, and J. Keshet, "Fooling end-to-end speaker verification with adversarial examples," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1962–1966.

[24] R. K. Das, X. Tian, T. Kinnunen, and H. Li, "The attacker's perspective on automatic speaker verification: An overview," *arXiv preprint arXiv:2004.08849*, 2020.

[25] H. Wu, S. Liu, H. Meng, and H.-y. Lee, "Defense against adversarial attacks on spoofing countermeasures of asv," *arXiv preprint arXiv:2003.03065*, 2020.

[26] S. Liu, H. Wu, H.-y. Lee, and H. Meng, "Adversarial attacks on spoofing countermeasures of automatic speaker verification," *arXiv preprint arXiv:1910.08716*, 2019.

[27] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[28] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," *arXiv preprint arXiv:1704.01155*, 2017.

[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pretraining of deep bidirectional transformers for language understanding," 2018.

[30] A. T. Liu, S.-w. Yang, P.-H. Chi, P.-c. Hsu, and H.-y. Lee, "Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020. [Online]. Available: http://dx.doi.org/10.1109/ICASSP40776.2020.9054458

[31] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018.

[32] Y.-A. Chung, W.-N. Hsu, H. Tang, and J. Glass, "An unsupervised autoregressive model for speech representation learning," *Interspeech 2019*, Sep 2019. [Online]. Available: http://dx.doi.org/10.21437/interspeech.2019-1473

[33] W. Wang, Q. Tang, and K. Livescu, "Unsupervised pre-training of bidirectional speech encoders via masked reconstruction," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020. [Online]. Available: http://dx.doi.org/10.1109/icassp40776.2020.9053541

[34] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[36] 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.