

# Adversarial Transformation of Spoofing Attacks for Voice Biometrics

Alejandro Gomez-Alanis, Jose A. Gonzalez-Lopez, Antonio M. Peinado

University of Granada

agomezalanis@ugr.es, joseangl@ugr.es, amp@ugr.es

## Abstract

Voice biometric systems based on automatic speaker verification (ASV) are exposed to *spoofing* attacks which may compromise their security. To increase the robustness against such attacks, anti-spoofing or presentation attack detection (PAD) systems have been proposed for the detection of replay, synthesis and voice conversion based attacks. Recently, the scientific community has shown that PAD systems are also vulnerable to adversarial attacks. However, to the best of our knowledge, no previous work have studied the robustness of full voice biometrics systems (ASV + PAD) to these new types of adversarial *spoofing* attacks. In this work, we develop a new adversarial biometrics transformation network (ABTN) which jointly processes the loss of the PAD and ASV systems in order to generate white-box and black-box adversarial *spoofing* attacks. The core idea of this system is to generate adversarial *spoofing* attacks which are able to fool the PAD system without being detected by the ASV system. The experiments were carried out on the ASVspoof 2019 corpus, including both logical access (LA) and physical access (PA) scenarios. The experimental results show that the proposed ABTN clearly outperforms some well-known adversarial techniques in both white-box and black-box attack scenarios.

**Index Terms:** Adversarial attacks, automatic speaker verification (ASV), presentation attack detection (PAD), voice biometrics.

## 1. Introduction

Voice biometrics aims to authenticate the identity claimed by a given individual based on the speech samples measured from his/her voice. Automatic speaker verification (ASV) [1] is the conventional way to put voice biometrics into practical usage. However, in recent years, ASV technology has been shown to be at risk of security threats performed by impostors who want to gain fraudulent access by presenting speech resembling the voice of a legitimate user [2, 3]. Impostors could use either logical access (LA) attacks [4], such as text-to-speech synthesis (TTS) and voice conversion (VC) based attacks, or physical access (PA) attacks such as replay based attacks [5].

To protect voice biometrics systems [6], it is common to develop anti-spoofing or presentation attack detection (PAD) [7] techniques which allow for differentiating between *bonafide* and *spoofing* speech [8, 9, 10]. Typically, the resulting biometrics system is a score-level cascaded integration of PAD and

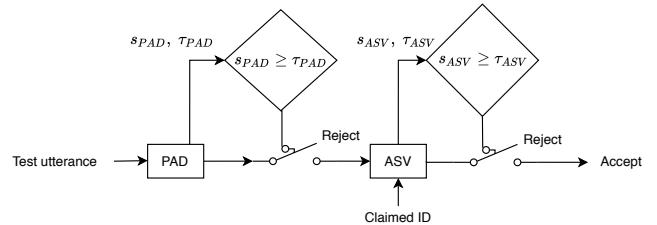


Figure 1: Block diagram of a score-level cascaded integration biometrics system.  $s_{PAD}$ ,  $\tau_{PAD}$  and  $s_{ASV}$ ,  $\tau_{ASV}$  denote the scores and thresholds of the PAD and ASV systems, respectively.

ASV subsystems, as depicted in Fig. 1. This is the same integration as the one used in the last two ASVspoof challenges [5, 11].

To make things more complex, different investigations [12, 13] have recently shown that PAD systems are also vulnerable to adversarial attacks [14]. These attacks can easily fool deep neural network (DNN) models by perturbing benign samples in a way normally imperceptible to humans [15]. Adversarial attacks can be divided into two main categories: white-box and black-box attacks. In this work, we refer to white-box attacks as those where the attacker can access all the information of the victim model (i.e., model architecture and its weights). Likewise, we will use the term black-box for those attacks where the attacker does not know any information about the victim model but it can be queried multiple times in order to estimate a surrogate model (student) of the victim model (teacher), using the binary responses (acceptance/rejection) of the victim model as ground-truth labels.

The main contributions of this work are:

- Investigate the robustness of full voice biometrics systems (ASV + PAD) under the presence of adversarial *spoofing* attacks.
- Propose an adversarial biometrics transformation network (ABTN) which is able to generate adversarial *spoofing* attacks in order to fool the PAD system without being detected by the ASV system.
- To the best of our knowledge, adversarial *spoofing* attacks have only been studied on logical access scenarios (TTS and VC based attacks). In this work, we also include physical access scenarios (replay based attacks).

The rest of this paper is organized as follows. Section 2 outlines some well-known adversarial attacks employed as baselines in this work. Then, in Section 3, we describe the proposed ABTN for white-box and black-box scenarios. After that, Section 4 outlines the speech corpora, systems details, and metrics employed in the experiments. Section 5 discusses the experimental results. Finally, we summarize the conclusions derived from this research in Section 6.

This work was supported in part by the Spanish State Research Agency (SRA) under grant PID2019-104206GB-I00/SRA/10.13039/501100011033. Alejandro Gomez-Alanis holds a FPU fellowship from the Spanish Ministry of Education (FPU16/05490). Jose A. Gonzalez-Lopez holds a Juan de la Cierva-Incorporación fellowship from the Spanish Ministry of Science, Innovation and Universities (IJC1-2017-32926). We also acknowledge the support of NVIDIA Corporation with the donation of a Titan X GPU.

## 2. Related Work

Adversarial *spoofing* examples can be generated by adding a minimally perceptible perturbation to the input *spoofing* utterance in order to do a refinement of the *spoofing* attack. In this work, we focus on targeted attacks, which aim to fool the PAD system by maximizing the probability of a targeted class (*bonafide*) different from the correct class (*spoof*). Specifically, to generate adversarial *spoofing* attacks, we fix the parameters  $\theta$  of a well-trained DNN-based PAD model and perform gradient descent to update the *spoofing* spectra of the input utterance so that the PAD model classifies it as a *bonafide* utterance. Mathematically, our goal is to find a sufficiently small perturbation  $\delta$  which satisfies:

$$\begin{aligned} \tilde{\mathbf{X}} &= \mathbf{X} + \delta, \\ f_{\theta}(\mathbf{X}) &= y, \\ f_{\theta}(\tilde{\mathbf{X}}) &= \tilde{y}, \end{aligned} \quad (1)$$

where  $f$  is a well-trained DNN-based PAD model parameterized by  $\theta$ ,  $\mathbf{X}$  denotes the sequence of speech feature vectors extracted from the input *spoofing* utterance (short time Fourier transform (STFT), typically),  $y$  is the true label corresponding to  $\mathbf{X}$ ,  $\tilde{y}$  is the targeted label class of the attack (*bonafide* class),  $\tilde{\mathbf{X}}$  denotes the perturbed input features, and  $\delta$  is the additive perturbation. Typically,  $\Delta$  is the feasible set of the allowed perturbation  $\delta$  ( $\delta \in \Delta$ ), which formalizes the manipulative power of the adversarial attack. Normally,  $\Delta$  is a small  $l_{\infty}$ -norm ball, that is,  $\Delta = \{\delta \mid \|\delta\|_{\infty} \leq \epsilon\}$ ,  $\epsilon \geq 0 \in \mathbb{R}$ .

There are multiple ways to generate the perturbation  $\delta$ , where the fast gradient sign method (FGSM) [16] and the projected gradient descent (PGD) [17] methods are the most popular adversarial attack procedures. The FGSM attack consists of taking a single step along the direction of the gradient, i.e.,

$$\delta = \epsilon \cdot \text{sign}(\nabla_{\mathbf{X}} \text{Loss}(\theta, \mathbf{X}, y)), \quad (2)$$

where  $\text{Loss}$  denotes the loss function of the neural network ( $\theta$ ), and the sign method simply takes the sign of its gradient. Unlike the FGSM, which is a single-step method, the PGD is an iterative method. Starting from the original input utterance  $\mathbf{X}_0 = \mathbf{X}$ , the input utterance is iteratively updated as follows:

$$\begin{aligned} \mathbf{X}_{n+1} &= \text{clip}(\mathbf{X}_n + \alpha \cdot \text{sign}(\nabla_{\mathbf{X}} \text{Loss}(\theta, \mathbf{X}, y)), \\ \text{for } n &= 0, \dots, N - 1, \end{aligned} \quad (3)$$

where  $n = 0, \dots, N - 1$  is the iteration index,  $N$  is the number of iterations,  $\alpha = \epsilon/N$ , and the  $\text{clip}()$  function applies element-wise clipping such that  $\|\mathbf{X}_n - \mathbf{X}\|_{\infty} \leq \epsilon$ ,  $\epsilon \geq 0 \in \mathbb{R}$ .

## 3. Proposed method

The performance of the FGSM and PGD methods are limited by the possibility of sticking at local optima of the loss function. Moreover, both methods have a limited search space ( $\Delta$ ) so that the perturbed *spoofing* speech  $\tilde{\mathbf{X}}$  is perceptually indistinguishable from the original *spoofing* speech  $\mathbf{X}$ .

In this work, we propose the Adversarial Biometrics Transformation Network (ABTN), which is a neural network that transforms a *spoofing* speech signal into an adversarial *spoofing* speech signal against a target biometrics system. Formally, an ABTN can be defined as a neural network  $g_{f,h} : \mathbf{X} \rightarrow \tilde{\mathbf{X}}$ , where  $f(\mathbf{X})$  and  $h(\mathbf{X})$  are the PAD and ASV models of the target biometrics system, respectively. The PAD and ASV models

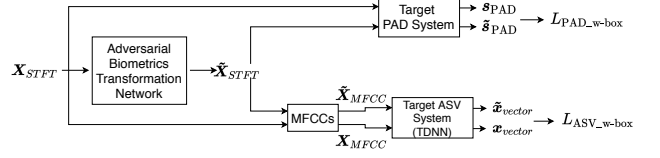


Figure 2: Proposed adversarial biometrics transformation system for white-box scenarios.

can provide either a probability distribution across class labels (white-box scenario) or just a binary decision (black-box scenario). In both scenarios, the objective of the ABTN is to generate adversarial *spoofing* attacks from *spoofing* speech in order to fool the PAD system while not being detected by the ASV system, i.e., while not modifying the speaker information.

### 3.1. White-box scenario

The architecture of the proposed ABTN system for the white-box scenario is depicted in Fig. 2. The output of the ABTN is fed into the target biometrics system which is composed of a PAD and an ASV system based on a time-delay neural network (TDNN) [18] for x-vector extraction (in fact, this is the only component of the ASV system that we need). The objective of this system is to train the ABTN so that it can generate adversarial attacks from *spoofing* speech which are able to fool the PAD system while, at the same time, it does not cause any changes to the ASV output (i.e., it does not change the speaker representation given by the corresponding x-vector). To train the ABTN, the PAD and ASV network parameters are frozen but the gradients are computed along them in order to back-propagate them to the ABTN parameters. To find the optimal parameters of the ABTN in the white-box (w-box) scenario, we minimize the following loss function:

$$L_{w\text{-box}} = L_{\text{PAD},w\text{-box}}(\mathbf{s}_{\text{PAD}}, \tilde{\mathbf{s}}_{\text{PAD}}) + \beta \cdot L_{\text{ASV},w\text{-box}}(\mathbf{x}_{\text{vector}}, \tilde{\mathbf{x}}_{\text{vector}}), \quad (4)$$

where,

$$L_{\text{PAD},w\text{-box}}(\mathbf{s}_{\text{PAD}}, \tilde{\mathbf{s}}_{\text{PAD}}) = \|r_{\alpha}(\mathbf{s}_{\text{PAD}}) - \tilde{\mathbf{s}}_{\text{PAD}}\|_2, \quad (5)$$

$$L_{\text{ASV},w\text{-box}}(\mathbf{x}_{\text{vector}}, \tilde{\mathbf{x}}_{\text{vector}}) = \|\mathbf{x}_{\text{vector}} - \tilde{\mathbf{x}}_{\text{vector}}\|_2. \quad (6)$$

$L_{\text{PAD},w\text{-box}}$  and  $L_{\text{ASV},w\text{-box}}$  are the loss components associated to the PAD and ASV systems, respectively, and  $\beta$  is a hyper-parameter to weight the importance of the two losses.  $\mathbf{s}_{\text{PAD}}$  and  $\tilde{\mathbf{s}}_{\text{PAD}}$  are the probability output vectors from the PAD system of the original and adversarial *spoofing* utterances, respectively. Likewise,  $\mathbf{x}_{\text{vector}}$  and  $\tilde{\mathbf{x}}_{\text{vector}}$  denote the x-vectors of the original and adversarial *spoofing* utterances, respectively, and  $r_{\alpha}$  is a reranking function which can be formulated as

$$r_{\alpha}(\mathbf{s}_{\text{PAD}}) = \text{norm} \left( \begin{cases} \alpha \cdot \max(\mathbf{s}_{\text{PAD}}) & k = 0 \\ \mathbf{s}_{\text{PAD}}(k) & k \neq 0 \end{cases} \right), \quad (7)$$

where  $k$  is the index class variable of the  $\mathbf{s}_{\text{PAD}}$  probability vector,  $\alpha > 1$  is an additional hyper-parameter which defines how large  $\mathbf{s}_{\text{PAD}}(k = 0)$ , i.e., the probability of the *bonafide* class, is with respect to the current maximum probability class, and  $\text{norm}$  is a normalizing function which rescales its input to be a valid probability distribution.

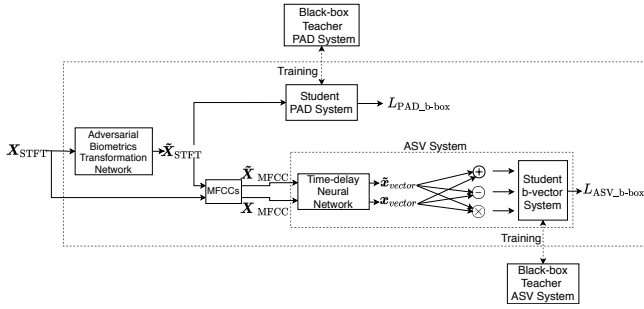


Figure 3: Proposed adversarial biometrics transformation system for black-box scenarios.

### 3.2. Black-box scenario

The architecture of the proposed ABTN system for the black-box scenario is depicted in Fig. 3. Similarly to the white-box scenario, the objective of this system is to generate adversarial attacks from *spoofing* speech which are able to fool the target (teacher) PAD system and, at the same time, bypass the target (teacher) ASV system by not modifying the speaker information represented by the corresponding x-vector. However, the limitation of the black-box scenario is that we do not have access to the parameters of the target biometrics system. Thus, we train a student PAD and a b-vector [19] based ASV systems by making requests to the target black-box biometrics system which only responds with a binary decision of acceptance or rejection, using these binary decisions as ground-truth labels. Therefore, the student PAD and b-vector systems are trained as binary classifiers in order to mimic the performance of the teacher PAD and ASV systems, respectively. Specifically, the student b-vector system computes the probability that the two input x-vectors belong to the same speaker, i.e., that  $P(b(\mathbf{x}_{\text{vector}}, \tilde{\mathbf{x}}_{\text{vector}}) = 1)$ , where  $b$  denotes the b-vector model.

To train the ABTN in the black-box scenario, the student PAD and ASV network parameters are also frozen but the gradients are computed along them in order to back-propagate them to the ABTN parameters. To find the optimal parameters of the ABTN in the black-box (b-box) scenario, we minimize the following loss function:

$$L_{\text{b-box}} = L_{\text{PAD,b-box}}(\tilde{\mathbf{s}}_{\text{PAD}}) + \beta \cdot L_{\text{ASV,b-box}}(\mathbf{x}_{\text{vector}}, \tilde{\mathbf{x}}_{\text{vector}}), \quad (8)$$

where,

$$L_{\text{PAD,b-box}}(\tilde{\mathbf{s}}_{\text{PAD}}) = \|\text{onehot}(k = 0) - \tilde{\mathbf{s}}_{\text{PAD}}\|_2, \quad (9)$$

$$L_{\text{ASV,b-box}}(\mathbf{x}_{\text{vector}}, \tilde{\mathbf{x}}_{\text{vector}}) = 1 - P(b(\mathbf{x}_{\text{vector}}, \tilde{\mathbf{x}}_{\text{vector}}) = 1). \quad (10)$$

$L_{\text{PAD,b-box}}$  and  $L_{\text{ASV,b-box}}$  are the loss components associated to the PAD and ASV systems, respectively. Moreover, the function *onehot* denotes the one-hot function and  $k = 0$  is the index of the *bonafide* class, so that the PAD system is fooled by firing the input *spoofing* utterance as a *bonafide* utterance.

## 4. Experimental Setup

This section briefly describes the speech corpora and metrics employed in our experiments, as well as the details of the proposed system.

### 4.1. Speech corpora

We conducted experiments on the ASVspoof 2019 database [20] which is split into two partitions for the assessment of LA and PA scenarios. This database also includes protocols for evaluating the performance of PAD, ASV and integration (biometrics) systems. Thus, we used this corpus for training the standalone PAD systems in the LA and PA scenarios, separately. Then, we generated adversarial *spoofing* attacks using only the *spoofing* utterances, so that they can bypass the biometrics system. We did not generate any adversarial examples from *bonafide* utterances, since we argue that they would not be *bonafide* anymore.

On the other hand, we also employed the Voxceleb1 [21] to train a TDNN [18] as an x-vector extractor for the ASV system. Also, following [6], a b-vector [19] ASV scoring system was trained in the black-box scenario using the *bonafide* utterances from the ASVspoof 2019 and Voxceleb1 development datasets.

### 4.2. Spectral analysis

Speech signals were analyzed using a Hanning analysis windows of 25 ms length with 10 ms of frame shift. Log-power magnitude spectrum features (STFT) with 256 frequency bins were obtained to feed all the PAD systems. The ASV systems were fed with Mel-frequency cepstral coefficients (MFCCs) obtained with the Kaldi recipe [22]. Only the first 600 frames of each utterance were used to extract acoustic features.

### 4.3. Implementation details

Two state-of-the-art PAD systems were adapted from different works, i.e., a light convolutional neural network (LCNN) [2] and a Squeeze-Excitation network (SENet50) [23]. The PAD scores were directly obtained from the *bonafide* class of the softmax output. For ASV, a TDNN x-vector model [18] was trained as an embedding extractor. Then, a probabilistic linear discriminant analysis (PLDA) [24] and a b-vector system [19] were trained as ASV scoring systems.

The proposed ABTN is formed by five convolutional layers with 16, 32, 48, 48 and 3 channels, respectively, and a kernel size of  $3 \times 3$ , followed by leaky ReLU activations. It was trained using the Adam optimizer [25] with a learning rate of  $3 \cdot 10^{-4}$ . Also, early stopping was applied to stop the training process when no improvement of the loss across the validation set was obtained. The values of  $\alpha$  and  $\beta$  were empirically set to 10 and 0.001, respectively, using a grid search on the validation set.

### 4.4. Evaluation setup

The PAD systems were evaluated using the pooled equal error rate ( $\text{EER}_{\text{spoof}}$ ) across all attacks. Likewise, the ASV systems were also evaluated using the  $\text{EER}_{\text{ASV}}$ , employing both *bonafide* utterances (target and non-target) and *spoofing* utterances. Any utterance rejected by either the PAD or ASV subsystems was assigned arbitrarily a  $-\infty$  score for computing the integration performance. Then, the integration (biometrics) systems were evaluated using the joint EER ( $\text{EER}_{\text{joint}}$ ) and the minimum normalized detection cost function (min-tDCF) [26] with the same configuration as the one employed in the ASVspoof 2019 challenge [11]. All the PAD, ASV and biometrics systems were evaluated using the ASVspoof 2019 test datasets.

System	Logical Access Attacks				Physical Access Attacks			
	EER <sub>spoofer</sub> (%)	EER <sub>ASV</sub> (%)	EER <sub>joint</sub> (%)	min-tDCF	EER <sub>spoofer</sub> (%)	EER <sub>ASV</sub> (%)	EER <sub>joint</sub> (%)	min-tDCF
No Attack	5.91	31.10*	20.13	0.1252	4.77	18.62*	13.37	0.1238
FGSM ( $\epsilon = 0.1$ )	5.98	31.14*	20.32	0.1279	7.50	18.65*	15.47	0.2157
PGD ( $\epsilon = 0.1$ )	5.95	31.13*	20.25	0.1267	6.08	18.63*	14.38	0.1717
FGSM ( $\epsilon = 1.0$ )	8.15	31.53*	25.44	0.1287	35.64	18.71*	26.54	0.9335
PGD ( $\epsilon = 1.0$ )	7.02	31.46*	25.37	0.1266	44.42	18.83*	26.77	0.9665
FGSM ( $\epsilon = 2.0$ )	2.01	30.11*	14.13	0.0623	1.02	17.61*	11.82	0.0380
PGD ( $\epsilon = 2.0$ )	4.97	31.38*	22.62	0.1078	29.29	18.44*	25.28	0.8677
FGSM ( $\epsilon = 5.0$ )	0.00	19.46*	2.45	0.0000	0.00	11.37*	11.79	0.0000
PGD ( $\epsilon = 5.0$ )	0.16	19.09*	2.56	0.0058	0.00	9.48*	11.79	0.0000
Proposed ABTN	<b>35.19</b>	<b>31.52*</b>	<b>39.15</b>	<b>0.5829</b>	<b>95.17</b>	<b>18.87*</b>	<b>36.63</b>	<b>1.0000</b>

Table 1: Results of the black-box adversarial attacks on the ASVspoof 2019 logical access (LA) and physical access (PA) test sets in terms of EER<sub>spoofer</sub>(%), EER<sub>ASV</sub>(%), EER<sub>joint</sub>(%) and min-tDCF. The target PAD system is based on a LCNN, while the student PAD system is based on a SENet50. The target ASV system is based on a TDNN + PLDA, while the student ASV system is based on a TDNN + b-vector. (\* ) The ASV evaluation includes both bonafide and spoofing utterances.

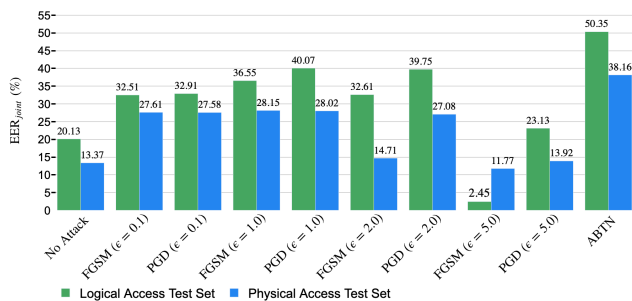


Figure 4: EER<sub>joint</sub>(%) of the white-box adversarial attacks on the ASVspoof 2019 logical and physical access test sets.

## 5. Experimental Results

The performance of the baseline biometrics system is shown in Table 1 as 'No Attack'. The LA and PA PAD systems are among the best single systems evaluated in the ASVspoof 2019 challenge [11]. The ASV system yields an EER of 4.75 and 7.25% in the LA and PA datasets when evaluating only the target and non-target *bonafide* utterances. However, its performance is degraded to 31.10 and 18.62% in the LA and PA test datasets when the *spoofing* utterances are also evaluated, as shown in Table 1.

### 5.1. White-box scenario

Fig. 4 shows the EER<sub>joint</sub> of the white-box adversarial attacks evaluated in the ASVspoof 2019 LA and PA test sets. The PAD and ASV systems are the state-of-the-art LCNN and TDNN + PLDA, respectively. As it was expected, PGD achieves slightly better results than FGSM due to its iterative procedure for generating the adversarial attacks. Moreover, the proposed ABTN outperforms the rest of adversarial attacks, obtaining 10.28% and 10.14% higher EER<sub>joint</sub> with respect to the best PGD configuration ( $\epsilon = 1.0$ ) in the LA and PA test sets, respectively. It is worth noticing that when the hyper-parameter  $\epsilon$  of the FGSM and PGD methods is equal or higher than 2.0, the biometrics system is able to detect the perturbation noise added by these adversarial attacks. In these cases, the performance of the *spoofing* attacks is even worse than when not using any adversarial attack (denoted by 'No Attack').

### 5.2. Black-box scenario

Table 1 shows the performance metrics for the black-box scenario. The target biometrics system consists of the same state-of-the-art LCNN (PAD) and TDNN + PLDA (ASV) systems evaluated in the previous section. The student PAD and ASV systems are the SENet50 and the TDNN + b-vector systems, respectively.

The proposed ABTN attacks outperform the best FGSM and PGD configurations by 27.04 and 50.75% of EER<sub>spoofer</sub>, and by 13.71 and 9.86% of EER<sub>joint</sub>, respectively. Also, the min-tDCF metric, which shows the performance of the biometrics system on a different operating point with respect to the EER<sub>joint</sub> [26], is significantly higher for the proposed ABTN adversarial attacks. As in the white-box scenario, it is worth noticing that the best adversarial attacks do not affect the performance of the ASV system with respect to the baseline system, since the perturbation noise of these attacks is not detected by the ASV system. However, when the hyper-parameter  $\epsilon \geq 2.0$ , both the PAD and ASV systems are able to detect the perturbations added by the FGSM and PGD methods, and hence, the biometrics system performs even better than the baseline system (denoted by 'No Attack'). However, the proposed ABTN method does not suffer from this issue since it is trained so that the added perturbation noise does not modify the speaker information from the *spoofing* utterance.

## 6. Conclusion

In this work, we studied the robustness of state-of-the-art voice biometrics systems (ASV + PAD) under the presence of adversarial *spoofing* attacks. Moreover, we proposed an adversarial biometrics transformation network (ABTN) for both white-box and black-box scenarios which is able to generate adversarial *spoofing* attacks in order to fool the PAD system without being detected by the ASV system. Experimental results have shown that biometric systems are highly sensitive to adversarial *spoofing* attacks in both logical and physical access scenarios. Moreover, the proposed ABTN system clearly outperforms other popular adversarial attacks such as the FGSM and PGD methods in both white-box and black-box scenarios. In the future, we would like to use the generated adversarial attacks for adversarial training in order to make the biometrics system more robust against these attacks.

## 7. References

- [1] R. Naika, "An overview of automatic speaker verification system," in *Advances in Intelligent Systems and Computing*. New York, NY, USA: Springer, 2018, vol. 673.
- [2] A. Gomez-Alanis, J. A. Gonzalez-Lopez, and A. M. Peinado, "A kernel density estimation based loss function and its application to ASV-spoofing detection," *IEEE Access*, vol. 8, pp. 108 530–108 543, 2020.
- [3] A. Gomez-Alanis, A. M. Peinado, J. A. Gonzalez, and A. M. Gomez, "A gated recurrent convolutional neural network for robust spoofing detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 1985–1999, 2019.
- [4] Z. Wu, T. Kinnunen, N. W. D. Evans, J. Yamagishi, C. Haniłçi, M. Sahidullah, and A. Sizov, "ASVspoof 2015: The first automatic speaker verification spoofing and countermeasures challenge," in *Proc. Interspeech*, Dresden, Germany, 2015, pp. 2037–2041.
- [5] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. W. D. Evans, J. Yamagishi, and K.-A. Lee, "The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. Interspeech*, Stockholm, Sweden, 2017, pp. 2–6.
- [6] A. Gomez-Alanis, J. A. Gonzalez-Lopez, S. P. Dubagunta, A. M. Peinado, and M. Magimai-Doss, "On joint optimization of automatic speaker verification and anti-spoofing in the embedding space," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1579–1593, 2021.
- [7] "Presentation attack detection." [Online]. Available: <https://www.iso.org/standard/67381.html>
- [8] A. Gomez-Alanis, A. M. Peinado, J. A. Gonzalez, and A. M. Gomez, "A deep identity representation for noise robust spoofing detection," in *Proc. Interspeech*, Hyderabad, India, 2018, pp. 676–680.
- [9] A. Gomez-Alanis, A. M. Peinado, J. A. Gonzalez, and A. M. Gomez, "Performance evaluation of front- and back-end techniques for asv spoofing detection systems based on deep features," in *Proc. Iberspeech*, Barcelona, Spain, 2018, pp. 45–49.
- [10] A. Gomez-Alanis, A. M. Peinado, J. A. Gonzalez, and A. M. Gomez, "A light convolutional GRU-RNN deep feature extractor for ASV spoofing detection," in *Proc. Interspeech*, Graz, Austria, 2019, pp. 1068–1072.
- [11] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. Kinnunen, and K. Lee, "ASVspoof 2019: Future horizons in spoofed and fake audio detection," in *Proc. Interspeech*, Graz, Austria, 2019, pp. 1008–1012.
- [12] S. Liu, H. Wu, H. yi Lee, and H. Meng, "Adversarial attacks on spoofing countermeasures of automatic speaker verification," in *Proc. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 312–319.
- [13] Y. Zhang, Z. Jiang, J. Villalba, and N. Dehak, "Black-box attacks on spoofing countermeasures using transferability of adversarial examples," in *Proc. Interspeech*, 2020, pp. 4238–4242.
- [14] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346 – 360, 2020.
- [15] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *Proc. International Conference on Learning Representations (ICLR)*, Banf, Alberta, Canada, 2014.
- [16] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.
- [17] Y. Deng and L. J. Karam, "Universal adversarial attack via enhanced projected gradient descent," in *Proc. IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 1241–1245.
- [18] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Alberta, Canada, 2018, pp. 5329–5333.
- [19] H.-S. Lee, Y. Tso, Y.-F. Chang, H.-M. Wang, and S.-K. Jeng, "Speaker verification using kernel-based binary classifiers with binary operation derived features," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, 2014, pp. 1660–1664.
- [20] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, and N. E. et al., "ASVspoof 2019: a large-scale public database of synthesized, converted and replayed speech," *Computer Speech and Language*, p. 101114, 2020.
- [21] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. Interspeech*, Stockholm, Sweden, 2017, pp. 2616–2620.
- [22] "SRE16 xvector model." [Online]. Available: <http://kaldi-asr.org/models/m3>
- [23] C.-I. Lai, N. Chen, J. Villalba, and N. Dehak, "ASSERT: Anti-Spoofing with Squeeze-Excitation and Residual Networks," in *Proc. Interspeech*, 2019, pp. 1013–1017.
- [24] S. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, 2007, pp. 1–8.
- [25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, 2015.
- [26] T. Kinnunen, K. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. Reynolds, "t-DCF: A detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification," in *Proc. Odyssey*, Les Sables d’Olonne, France, 2018, pp. 312–319.