Signal Processing Interpretation for Adversarial Examples in Speaker Verification

1st Sreekanth Sankala
Speech Information Processing Lab
Department of Electrical Engineering
IIT Hyderabad
ee20resch11011@iith.ac.in

2nd Sri Rama Murty Kodukula
Speech Information Processing Lab
Department of Electrical Engineering
IIT Hyderabad
ksrm@ee.iith.ac.in

3rd Yegnanarayana B
Speech Information Processing Lab
Department of Electrical Engineering
IIT Hyderabad
yegna@ee.iith.ac.in

***Abstract*—To safeguard automatic speaker verification (ASV) systems due to attacks from adversarial examples, it is important to understand the characteristics of the adversarial noise, and its effect on the speech signal. This study aims at explaining the effect of adversarial noise on signal characteristics in the spectral domain. The adversarial examples seem to exploit the low energy regions in the time-frequency (TF) representation of the test speech signal to alter the decision of the ASV system. It appears that the phase of the adversarial noise plays a role in increasing the spectral dynamic range of the test speech signal, resulting in peaks and valleys in the spectral envelope in the low-energy frequency components. Based on these new insights, we explain the reasons behind the success of the existing preprocessing defense strategies. This work may help researchers to develop more intuitive defense strategies.**

***Index Terms*—speech signal analysis, speaker recognition, adversarial attacks, spectral dynamic range, defense strategies**

I. Introduction

Automatic speaker verification (ASV) system authenticates the claimed identity of a speaker from his/her voice characteristics. Deployment of ASV systems in security-critical applications necessitates careful investigations of possible attacks against them [1]. Recent experimental studies exposed the vulnerability of ASV systems to adversarial attacks [2]–[4]. An adversarial attack against the ASV system involves slightly perturbing the test speech signal (s) to overturn the verification result of ASV system, i.e., accept a different speaker trial or reject the same speaker trial. In general, the perturbed test speech signal is indistinguishable from the original test speech signal by the humans, but its effect is changing the decision by the model. These perturbed inputs are named as adversarial examples. While the initial studies have established the existence of adversarial examples for the end-to-end ASV systems [2], the research community quickly developed algorithms to attack the state-of-art ASV systems [3], [4]. Some studies even demonstrated the feasibility of real-life attacks on ASV systems via black-box, over-the-air, and universal adversarial attacks [5]–[7].

In response to the attacks, numerous defense strategies have been proposed in the literature to secure the ASV system from such attacks. There are two broad categories of defense strategies against adversarial attacks: 1) Proactive defense strategies and 2) Reactive defense strategies. Proactive defense strategies primarily operate at the model level and typically require retraining the model to make it inherently robust to adversarial examples [8], [9]. In contrast, reactive defense strategies involve preprocessing the input data before feeding it to the model. The preprocessing operation aims to reduce the adversarial features in the signal to act as a defense technique. Recently, the preprocessing defense strategies received more attention in the ASV community, and several techniques were proposed to safeguard the ASV system from adversarial attacks [10]–[12]. Some studies utilized the preprocessing operations to detect adversarial examples in the test data, and excludes them from the evaluation itself [13]–[16].

Although many attacks and their defense strategies have been proposed in the literature, most of these works fail to provide a satisfactory explanation of the characteristics of adversarial noise and its effect on the speech signal. Understanding the impact of adversarial noise on the characteristics of the signal offers better insights in devising robust defense techniques. In contrast to the earlier works that aim to attack the system or defend against the attacks, this work mainly focuses on explaining the effects of adversarial noise on the signal characteristics in the spectral domain. Our studies show that the adversarial noise predominantly exploits the low-energy regions in the time-frequency representation of the signal. We demonstrate that the addition of adversarial noise to the signal increases the dynamic range of the magnitude spectrum. Further analysis revealed that the phase of adversarial noise plays a crucial role in increasing the dynamic range of the magnitude spectrum. Specifically, we observe that synchronizing the adversarial noise phase with the signal phase increases the dynamic range of the magnitude spectrum. Experiments were conducted to support our assertions. Finally, our observations are utilized to justify the working of the preprocessing defense strategies.

The rest of the paper is organized as follows. Section II details the ASV system and the corresponding adversarial attacks against it. Section III describes how the addition of adversarial noise increases the dynamic range of the magnitude spectrum. Section IV demonstrates the importance of the phase of adversarial noise in attacking the system. Experimental details, results, and analysis are included in Section V. Finally, Section VI summarizes the contributions of the paper.

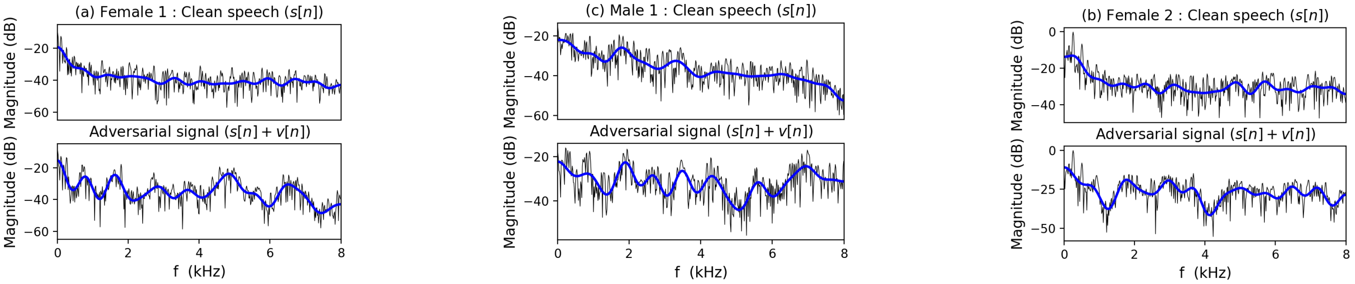

Fig. 1: (black) Magnitude spectrum of the low energy segment (silence frame) of the clean, adversarial signals from three different speakers (2 female + 1 male). (blue) Spectral envelope computed from the first 23 cepstral coefficients of the spectrum. The adversarial noise is added to the signal at an approximate SNR of 33 dB.

II. Adversarial Attacks Against ASV

The objective of ASV system is to verify whether the enrollment ($e[n]$) and test speech signals ($s[n]$) are spoken by the same speaker. An adversarial attack against the ASV system involves perturbing the test speech signal ($s[n]$) to alter the decision of the ASV system. While the literature includes several methods to generate adversarial examples, this work focuses on additive white-box gradient-based adversarial attacks. These attacks assume complete knowledge about the model and exploits the gradient information to generate the adversarial noise , and adds it to the test signal to generate the adversarial example. The working principle of these algorithms can be explained by examining the first-order Taylor series approximation of the adversarial score at clean score

$$\begin{aligned} a[n] &= s[n] + v[n] \\ f_{\mathbf{w}}(e[n], a[n]) &= f_{\mathbf{w}}(e[n], s[n] + v[n]) \\ &\approx f_{\mathbf{w}}(e[n], s[n]) + v^T[n] \nabla_{s[n]} f_{\mathbf{w}}(e[n], s[n]) \end{aligned} \tag{1}$$

where $e[n]$ and $s[n]$, are the enrollment and clean test signals. $v[n]$ is the adversarial noise added to $s[n]$ to create the adversarial signal $a[n]$. $f_{\mathbf{w}}(e[n], s[n])$ is the speaker similarity score between enrollment and clean test utterances, computed by the ASV system $f_{\mathbf{w}}(.,.)$ parameterized by $\mathbf{w}$. From the Taylor series approximation, it can be seen that, for the added adversarial noise $v[n]$ to be smaller yet able to increase the deviation between the clean and adversarial score, it needs to be in correlation with the gradients of the score function with respect to the input. Gradient-based white-box attacks explores this idea and adds the noise which is correlated with the input gradients of the test signal to generate the adversarial signal. This work studies the vulnerability of ASV system to the basic iterative method (BIM) [14], [17]. It is an iterative version of the fast gradient sign method (FGSM) [3], [18]. In the BIM attack, the attacker will start from $a^1[n] = s[n]$ and iteratively updates it to find the adversarial signal. The k^{th} iterative update of the BIM attack is given as

$$\begin{aligned} v^k[n] &= (-1)^{is\_tgt} \cdot sign(\nabla_{a^k[n]} f_{\mathbf{w}}(e[n], a^k[n])) \\ a^{k+1}[n] &= Clip_\epsilon(a^k[n] + \alpha \cdot v^k[n]) \end{aligned} \tag{2}$$

where $Clip_\epsilon(.)$ is the clipping function which make sure that $||a^{k+1}[n] - s[n]||_\infty \leq \epsilon$, and ϵ is the maximum allowed perturbation in the BIM attack. α is the step size that quantifies the amount of perturbation in each iteration. is_tgt = 1 and -1 for the genuine and imposter trails, respectively. This procedure is repeated for N iterations to generate the adversarial example ($a[n]$). More details about the BIM attack against ASV can be found in [13].

III. Analysis of Adversarial Noise

It is observed that adversarial noise added at an approximate signal-to-noise (SNR) of 30 dB remains imperceptible to humans, but it can lead to misclassification by the ASV system. We examine the characteristics of the signal that are effected by this adversarial noise.

A. Effect of Adversarial Noise on Signal

Since the adversarial noise is added at high SNR to the speech signal, it is expected to effect silence and speech regions dissimilarly. Hence, to begin with, we have computed the magnitude spectrum of low-energy segment of utterances from three different speakers and presented in Figure 1. It shows that the addition of adversarial noise to the signal changes its magnitude spectrum. It introduces peaks and valleys to the spectral envelope, as indicated by the cepstrally smoothed spectra in Figure 1. This changes in the magnitude spectrum comes from the addition of adversarial noise whose phase is correlated with the signals phase. The magnitude spectrum of adversarial signal ($a[n]$) is given by

$$\begin{aligned} |A(j\omega)|^2 &= |S(j\omega)|^2 + |V(j\omega)|^2 \\ &\quad + 2|S(j\omega)||V(j\omega)|cos(\angle S(j\omega) - \angle V(j\omega)) \end{aligned} \tag{3}$$

From Eq (3), it is observed that adding adversarial noise to $s[n]$ modifies its magnitude spectrum through the inclusion of an error component comprising two terms. The first term of the error component involves the magnitude spectrum of adversarial noise, while the second involves the cosine of the phase differences. The adversarial noise is added to the signal at a higher signal-to-noise ratio (SNR) to be imperceptible to human beings. The magnitude spectrum of adversarial noise has a low value, and it is expected to introduce negligible changes to the magnitude spectrum of $\mathbf{s}$. In contrast, the second term of the error component involving the cosine of phase differences adds prominent features to the magnitude spectrum. This is explained as follows: When the cosine of the phase difference is considered, both positive and negative values increase the dynamic range of the magnitude spectrum. Consequently, the magnitude spectrum of adversarial noise requires higher values to achieve the same level of expansion of the dynamic range as the phase spectrum. Thus the adversary synchronizes the phase of the adversarial noise with the signal phase to increase the dynamic range of the spectrum and induces local peak-valley structure.

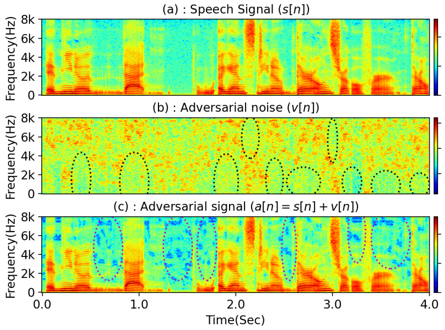

Fig. 2: Log magnitude spectrogram of the signals. The adversarial noise is added to the signal at 48 dB SNR.

To broaden the scope of our analysis to the entire speech signal, we have presented the log magnitude spectrogram of the speech signal, adversarial noise, and the corresponding adversarial signal in Figure 2. The log magnitude spectrograms of $v[n]$ and $a[n]$ indicates that the adversarial noise predominantly disturbs the low energy regions in the log magnitude spectrogram of $s[n]$. It mainly adds peaks and valleys to the spectral envelope associated with the low energy frequency components. This observation is highlighted with dotted ellipses in Figure 2. The presence of blue patches in the log magnitude spectrogram of the adversarial signal, indicates the valleys in the spectral envelope, and they are induced by the synchronization of the adversarial phase with the signal phase. In summary, this analysis reveals that the adversarial noise effects predominantly the low energy regions in the time-frequency representation of test speech signal, and the effect is mainly through the phase of adversarial noise. It is interesting to note that this observations justifies the success of the spectral masking-based preprocessing operation proposed in [14], [15].

B. Effect of Adversarial Noise on Model Outputs

In an effort to explain how the added adversarial features (local spectral peak valley structure) overturns the ASV system decision, we examined the effect of adversarial noise on the intermediate outputs of the ASV system. The ASV system in this work, projects the speech signal onto a speaker discriminative space, and compares the resultant projections, known as speaker embeddings, using the backend scoring modules such as cosine similarity, PLDA scoring, etc. In this work, the speaker embeddings are extracted by tapping the activation potentials of a penultimate layer of a speaker classification network trained with a huge number of background speakers. This work utilized the most widely used x-vector architecture to build the speaker classification network [19]. Several variants of the x-vector framework are proposed to improve the performance of the ASV system [19]–[24]. This study mainly utilizes the vanilla x-vector system for the analysis [19]. Further, the performance is evaluated using the state-of-the-art x-vector framework, i.e., Squeeze-and-Excitation ResNet (SEResNet), discussed in [20]. The vanilla x-vector architecture comprises three sub-modules: frame-level encoder, time-attentive statistical pooler, and speaker classifier. The frame-level encoder consists of 1D convolution layers to extract high-level representations ($\mathbf{H} \in \mathcal{R}^{T \times D}$) from the log Mel-filter bank energies ($\mathbf{X} \in \mathcal{R}^{T \times C}$) of the input speech signal. The time-attentive stats pooling layer aggregates the variable length representations to obtain a fixed dimensional embedding $[\boldsymbol{\mu}, \boldsymbol{\sigma}]$, computed as follows

$$\boldsymbol{\mu} = \sum_{t=1}^{T} \mathbf{h}_t w_t \qquad \boldsymbol{\sigma} = \sqrt{\sum_{t=1}^{T} diag((\mathbf{h}_t - \boldsymbol{\mu})(\mathbf{h}_t - \boldsymbol{\mu})^T) w_t} \tag{4}$$

where w_t indicates the relative contribution of frame level representation $\mathbf{h}_t$ towards the fixed dimensional embedding. Finally, the segment-level encoder comprising of couple of dense layers is operated on fixed dimensional embedding to estimate the speaker posteriors. The parameters of this classification network are optimized by minimizing the additive angular margin (AAM) softmax loss [25]. In-depth information about the vanilla x-vector and SEResNet x-vector based ASV systems can be found in [19] and [20], respectively. Once the speaker classification network is trained, the speaker embeddings are extracted for enrollment ($e[n]$) and test speech signals ($s[n]$), and compared using the cosine scoring function to obtain the speaker similarity score between them, i.e., $f_{\mathbf{w}}(e[n], s[n])$, where $\mathbf{w}$ represents the parameters of the classification network.

In order to show that the extracted speaker embedding gets effected by the added adversarial features in the input signal, we examine the effect of adversarial noise on frame-level representations and their relative contribution towards the speaker embedding. We analyzed the squared absolute errors induced at the frame-level representation by the addition of adversarial noise to the input signal.

$$f_a[t] = \frac{1}{D} \sum_{d=1}^{D} (\mathbf{h}_t^a - \mathbf{h}_t^s)^T (\mathbf{h}_t^a - \mathbf{h}_t^s) \quad \forall t = 1, 2, ..., T \tag{5}$$

Where $\mathbf{h}_t^s \in \mathcal{R}^D$, $\mathbf{h}_t^a \in \mathcal{R}^D$ denotes the t^{th} frame level representation of the clean signal and adversarial signal, respectively. The $f_a[t]$ indicates how much the t^{th} frame-level representation affected by the addition of adversarial noise to the input signal. Figure 3 displays the plots of squared absolute errors and the corresponding relative weights assigned to them. In Figure 3, the relative weights are scaled to match

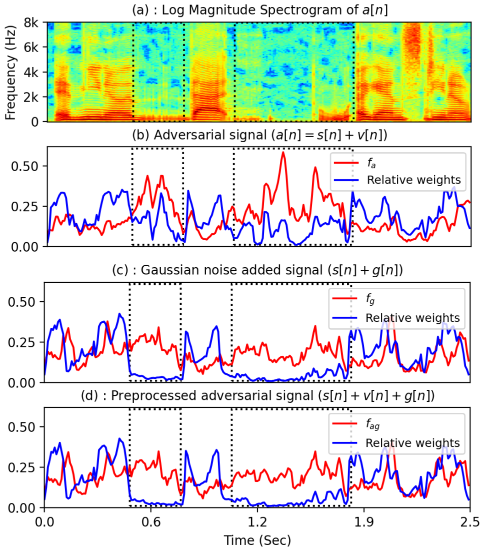

Fig. 3: Effect of adversarial noise on intermediate outputs of ASV system. $g[n]$ is Gaussian noise. f_{ag} refers to the absolute errors rendered by the preprocessed adversarial signal, i.e., the addition of Gaussian noise to the adversarial signal.

the range of absolute errors. From Figure 3(b), it is evident that the addition of adversarial noise manipulates the frame-level representations corresponding to the erroneous regions, i.e., signal regions containing the adversarial features which are manifested as blue patches in the log magnitude spectrogram. These erroneous regions and the changes in their frame-level representations are highlighted using the dotted rectangular boxes in Figures 3(a) and 3(b), respectively. The relative weights plot in Figure 3(a) indicates that the contribution of these manipulated frame-level representations is more towards the extraction of speaker embedding. As a result, the addition of adversarial noise pushes the adversarial speaker embedding far away from the clean speaker embedding, resulting in the misclassification of adversarial input.

In order to highlight the uniqueness of adversarial noise in attacking the system, we compare it with the added Gaussian noise, in Figure 3(c). The squared absolute errors in Figure 3(c) indicates that the overall effect is considerably lower in the Gaussian noise case compared to the adversarial noise. Similar to the adversarial noise case, even in the Gaussian noise case, a few of the frame-level representations are affected relatively more. However, they received relatively less weight in the attentive stats pooler, resulting in the nullification of their effect in the final speaker embedding. This analysis indicates that, it is the added spectral peaks and valleys that cause the misclassification of the input.

IV. Importance of Phase of Adversarial Noise in Attacking the System

In this section, we describe the experiment that validates our observation that the phase of adversarial noise plays a crucial role in increasing the dynamic range of the magnitude spectrum. We specifically assessed the efficacy of the adversarial noise magnitude and phase in attacking the system. We evaluated the magnitude and phase spectrogram of the adversarial noise. Then the phase intact adversarial noise is generated through Inverse Short Time Fourier Transform (ISTFT) operation on the complex spectra generated by combining the phase spectra of the adversarial noise with the magnitude spectra of Gaussian noise. Similarly, magnitude-intact adversarial noise is estimated from the magnitude spectra of adversarial noise and the phase spectra of Gaussian noise. Let **a** and **g** be the adversarial and Gaussian noises, respectively. Then the phase or magnitude intact adversarial noises can be generated as follows.

$$\begin{aligned} \hat{a}_{phase}[n] &= ISTFT\left[\left|G(j\omega)\right| e^{j\angle A(j\omega)}\right] \\ \hat{a}_{mag}[n] &= ISTFT\left[\left|A(j\omega)\right| e^{j\angle G(j\omega)}\right] \end{aligned} \tag{6}$$

where $\hat{a}_{phase}[n]$ and $\hat{a}_{mag}[n]$ are the phase and magnitude intact adversarial noises, respectively. After that, $\hat{a}[n]_{phase}$ or $\hat{a}[n]_{mag}$ are individually added to the speech signals to attack the system. Our analysis suggests that $\hat{a}[n]_{phase}$ is more effective in attacking the system compared to the $\hat{a}[n]_{mag}$. The experimental results reported in Section V-B agree with our observations, providing support for our argument.

V. Results & Analysis

This section describes the datasets, experimental details, results, and analysis of adversarial attacks against ASV systems.

A. Datasets & Experimental Details

The speaker classification networks are trained with VoxCeleb1 dev partition comprising 148k utterances from 1211 speakers [26]. A subset of VoxCeleb 1 test data [26] comprising 10k trials from 40 speakers, with an equal portion of target and non-target trials, is used to evaluate the adversarial vulnerability of ASV systems. Log Mel filter bank energies with 64 filters are used as input to the speaker classification network. In Additive Angular Margin (AAM) softmax loss, the margin and scale are chosen as 0.2 and 30 [25]. In the BIM adversarial attack, the step size (α) is selected as one-tenth of the maximum allowed perturbation (ϵ), and the number of iterations (N) is chosen as 30. The performance of the ASV system is quantified in terms of Equal Error Rate (EER).

B. Importance of phase of adversarial noise in attacking

Table I gives the performance of the ASV systems on clean and adversarial examples generated with different maximum allowed perturbation factors in the BIM attack. It shows that adding adversarial noise to test speech signal even at 48 dB completely overturns the verification result of the x-vector system, that is the performance degradation from 5.3 % EER to 92 % EER. It also demonstrates that even the state-of-the-art ASV system (SEResNet) is also vulnerable to adversarial attacks.

The last two rows of Table I shows the performance of the ASV system on the adversarial examples generated by adding

TABLE I: The table reports the EER of the ASV systems on clean and adversarial examples generated with different perturbation factors in the BIM attack. In the table, ϵ refers to the maximum allowed perturbation in the BIM attack. Results corresponding to $\epsilon = 0$ are the performance of the ASV system on clean data. The table also reports the average SNR at which the adversarial noises are added to the test speech signals.

System	x-vector			SEResNet		
ϵ x 10^3	0.0	0.5	4	0.0	0.5	4
SNR (dB)	∞	48	33	∞	48	33
Adversarial Vulnerability of ASV systems						
Adversarial noise $(a[n])$	5.3	92	99	3.5	95	99
Importance of phase of adversarial noise						
Impact of phase : $\hat{a}_{phase}[n]$	-	**75**	**99**	-	**85**	**99**
Impact of magnitude : $\hat{a}_{mag}[n]$	-	7.8	43	-	4.6	18

the magnitude or phase intact adversarial noise to the test speech signals. The evaluation with $\hat{a}_{phase}[n]$ indicates that the phase of the adversarial noise plays a crucial in attacking the system. The magnitude of adversarial noise is relatively less important in attacking the system, especially at high SNR attacks. While we arrived at similar conclusions with other attacks (FGSM), we could not include them in the paper due to space limitations. These experimental evaluation supports our signal processing based explanation given in Section III.

C. Justification of preprocessing defense strategies

TABLE II: The table reports the EER of the ASV systems on original and preprocessed adversarial signals. G(.), L(.), M(.), and N(.) refer to the following preprocessing operations - Gaussian noise addition, Linear spectrogram inversion, Mel spectrogram inversion, and Neural vocoder, respectively.

System	x-vector				SEResNet			
ϵ x 10^3	0.0	0.3	0.5	4	0.0	0.3	0.5	4
SNR	∞	52	48	33	∞	52	48	33
Evaluation with original adversarial signal								
$a[n]$	5.3	77	92	99	3.5	83	95	99
Evaluation with preprocessed adversarial signals								
$G(a[n])$	7.9	10	12	56	4.3	5.5	6.4	37
$L(a[n])$	5.6	30	53	99	4.9	24	44	99
$M(a[n])$	5.6	20	35	99	4.8	14	25	96
$N(a[n])$	5.7	44	65	99	4.0	21	37	93

Typically, the preprocessing defense modifies the input before feeding it to the model to eliminate the effect of adversarial noise in the signal. Various preprocessing operations have been proposed in the literature to defend against adversarial attacks. However, most of the works failed to provide a satisfactory explanation for their success. This work justifies the working of the following four preprocessing defense strategies using our spectral analysis.

- Gaussian noise addition $(G(a[n]))$ [12]: Gaussian noise is added to the adversarial signal to reduce the effectiveness of the features of adversarial noise in the signal.
- Linear spectrogram inversion $(L(a[n]))$ [13]: This preprocessing operation reconstructs the speech signal from the magnitude spectrogram combined with the phase spectrogram estimated using the Griffin Lim algorithm.
- Mel spectrogram inversion $(M(a[n]))$ [13]: It reconstructs the speech signal from Mel spectrogram.
- Neural vocoder $(N(a[n]))$ [13]: Employs neural vocoder [27] to project the adversarial signal onto the basis of the clean signal.

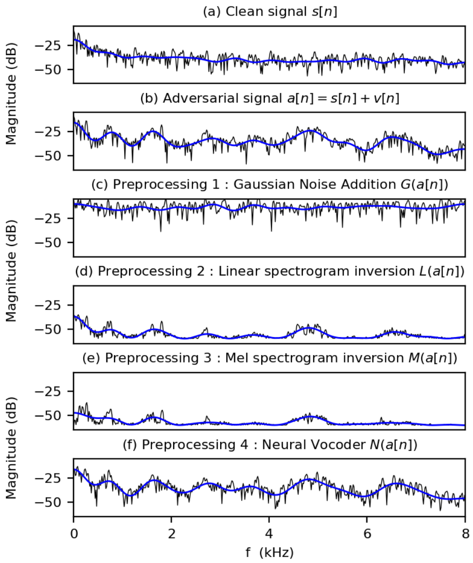

Fig. 4: (black) Magnitude spectrum. (blue) Spectral envelope computed from its first 23 cepstral coefficients.

Figure 4 shows the magnitude spectrum of the low energy segment (silence frame) of clean, adversarial, and four preprocessed adversarial signals. From Figure 1(c), it is observed that the preprocessing operation involving the addition of Gaussian noise to the adversarial signal $(G(a[n]))$, adds bias to the magnitude spectrum. As a result, this preprocessing operation decreases the dynamic range of the magnitude spectrum, and nullifies the spectral peaks and valleys introduced by the adversarial noise. On the other hand, Figures 1(d) and (e) show that Linear or Mel spectrogram inversions hold the features of the adversarial noise in the preprocessed signal. Figure 1(f) demonstrates that the neural vocoder as a preprocessing operation clearly preserves the added spectral peaks and valleys, compared to all the other preprocessing operations. As a result, the Gaussian noise addition is expected to perform well as a preprocessing defense, and the neural vocoders are expected to be less effective as a preprocessing defense than others.

We have evaluated the performances of the ASV system on the preprocessed signals and presented in Table II. The preprocessing operation is said to be effective if the EER of the ASV system evaluated on the preprocessed signals is closer to the clean signal EER. Table II shows that the Gaussian noise addition as a preprocessing defense outperforms the other preprocessing defenses. The preprocessing operation that uses the neural vocoder is shown to be less effective in combating adversarial attacks. These results support our observation that adding adversarial noise increases the dynamic range of the magnitude spectrum, thereby introducing peaks and valleys in the spectral envelope. The model leverages these added spectral peaks and valleys to misclassify the input. The preprocessing operations that reduce the added spectral features are expected to perform well in defending against adversarial attacks. Gaussian noise addition, which is shown to remove the adversarial features, outperforms remaining defense strategies. Similar conclusions are deduced from the effect of preprocessed adversarial signal on model outputs, presented in Figure 3(d). It shows that the addition of Gaussian noise to the adversarial signal reduces the squared absolute errors at frame-level representations, and avoids the misclassification of input.

VI. Conclusions

In this work, we attempted to provide a signal-processing interpretation for adversarial examples in speaker verification systems. We observe that the adversarial example exploits the low-energy regions in the time-frequency representation of the test speech signal to alter the ASV system decision. The phase of adversarial noise plays a crucial role in increasing the dynamic range of the magnitude spectrum, thereby introducing the peaks and valleys in the spectral envelope. The attention modules are explored to show that the model responds to the added adversarial features such as local peak valley structure in the spectrum to misclassify the input. Finally, our observations are related to the existing preprocessing defense strategies to assess their potential. One of the extensions of this work could be development of defense strategies that aim to eliminate the added peaks and valleys in the spectral envelope.

VII. ACKNOWLEDGMENT

This work was supported by DST National Mission Interdisciplinary Cyber-Physical Systems (NM-ICPS), Technology Innovation Hub on Autonomous Navigation and Data Acquisition Systems: TiHAN Foundations at Indian Institute of Technology (IIT) Hyderabad

References

[1] IDVoice, "Innovative voice verification software from id r&d." in *https://www.idrnd.ai/voice-biometrics/.*

[2] 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.

[3] X. Li, J. Zhong, X. Wu, J. Yu, X. Liu, and H. Meng, "Adversarial attacks on gmm i-vector based speaker verification systems," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6579–6583.

[4] J. Villalba, Y. Zhang, and N. Dehak, "x-vectors meet adversarial attacks: Benchmarking adversarial robustness in speaker verification." in *INTERSPEECH*, 2020, pp. 4233–4237.

[5] Z. Li, C. Shi, Y. Xie, J. Liu, B. Yuan, and Y. Chen, "Practical adversarial attacks against speaker recognition systems," in *Proceedings of the 21st international workshop on mobile computing systems and applications*, 2020, pp. 9–14.

[6] G. Chen, S. Chenb, L. Fan, X. Du, Z. Zhao, F. Song, and Y. Liu, "Who is real bob? adversarial attacks on speaker recognition systems," in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021, pp. 694–711.

[7] Y. Xie, C. Shi, Z. Li, J. Liu, Y. Chen, and B. Yuan, "Real-time, universal, and robust adversarial attacks against speaker recognition systems," in *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2020, pp. 1738–1742.

[8] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein, "Adversarial training for free!" *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[9] M. Pal, A. Jati, R. Peri, C.-C. Hsu, W. AbdAlmageed, and S. Narayanan, "Adversarial defense for deep speaker recognition using hybrid adversarial training," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6164–6168.

[10] H. Wu, X. Li, A. T. Liu, Z. Wu, H. Meng, and H.-Y. Lee, "Improving the adversarial robustness for speaker verification by self-supervised learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 202–217, 2021.

[11] H. Wu, Y. Zhang, Z. Wu, D. Wang, and H.-y. Lee, "Voting for the right answer: Adversarial defense for speaker verification," *arXiv preprint arXiv:2106.07868*, 2021.

[12] L.-C. Chang, Z. Chen, C. Chen, G. Wang, and Z. Bi, "Defending against adversarial attacks in speaker verification systems," in *2021 IEEE International Performance, Computing, and Communications Conference (IPCCC)*. IEEE, 2021, pp. 1–8.

[13] H. Wu, P.-c. Hsu, J. Gao, S. Zhang, S. Huang, J. Kang, Z. Wu, H. Meng, and H.-y. Lee, "Adversarial sample detection for speaker verification by neural vocoders," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 236–240.

[14] X. Chen, J. Yao, and X.-L. Zhang, "Masking speech feature to detect adversarial examples for speaker verification," in *2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2022, pp. 191–195.

[15] X. Chen, J. Wang, X.-L. Zhang, W.-Q. Zhang, and K. Yang, "Lmd: A learnable mask network to detect adversarial examples for speaker verification," *arXiv preprint arXiv:2211.00825*, 2022.

[16] Z. Chen, "On the detection of adaptive adversarial attacks in speaker verification systems," *arXiv preprint arXiv:2202.05725*, 2022.

[17] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*. Chapman and Hall/CRC, 2018, pp. 99–112.

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

[19] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification." in *Interspeech*, vol. 2018, 2018, pp. 3573–3577.

[20] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," *arXiv preprint arXiv:2003.11982*, 2020.

[21] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[22] Y.-Q. Yu and W.-J. Li, "Densely connected time delay neural network for speaker verification." in *INTERSPEECH*, 2020, pp. 921–925.

[23] J. Villalba, N. Chen, D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, J. Borgstrom, F. Richardson, S. Shon, F. Grondin *et al.*, "State-of-the-art speaker recognition for telephone and video speech: The jhu-mit submission for nist sre18." in *Interspeech*, 2019, pp. 1488–1492.

[24] Z. Bai and X.-L. Zhang, "Speaker recognition based on deep learning: An overview," *Neural Networks*, vol. 140, pp. 65–99, 2021.

[25] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.

[26] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[27] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6199–6203.