Phase perturbation improves channel robustness for speech spoofing countermeasures

Yongyi Zang, You Zhang, Zhiyao Duan

University of Rochester, Rochester, New York, USA

{yongyi.zang, you.zhang, zhiyao.duan}@rochester.edu

Abstract

In this paper, we aim to address the problem of channel robustness in speech countermeasure (CM) systems, which are used to distinguish synthetic speech from human natural speech. On the basis of two hypotheses, we suggest an approach for perturbing phase information during the training of time-domain CM systems. Communication networks often employ lossy compression codec that encodes only magnitude information, therefore heavily altering phase information. Also, state-of-the-art CM systems rely on phase information to identify spoofed speech. Thus, we believe the information loss in the phase domain induced by lossy compression codec degrades the performance of the unseen channel. We first establish the dependence of time-domain CM systems on phase information by perturbing phase in evaluation, showing strong degradation. Then, we demonstrated that perturbing phase during training leads to a significant performance improvement, whereas perturbing magnitude leads to further degradation.

Index Terms: speech recognition, human-computer interaction, computational paralinguistics

1. Introduction

Speech generative systems have been progressing rapidly in recent years [1, 2]. The state-of-the-art speech generative model VALL-E [3] can even mimic a person's voice with only 3 seconds of speech. If misused by criminals, these deep generative algorithms could aid in spoofing attacks. Therefore, the research community has been developing speech countermeasure (CM) systems for distinguishing synthetic speech from human natural speech. For real-world applications, spoofing attacks could be generated by algorithms and transmitted through communication channels that are novel to the CM system. Therefore, CM systems need to generalize to unseen synthetic attacks and channel variations [4].

Such generalization ability has been studied by the CM community, especially driven by the ASVspoof challenge series [5]. In ASVspoof2019, the spoofing attacks in the evaluation set are created with different generative algorithms than the training set, enforcing the evaluation on the generalization ability to unseen attacks. Raw-waveform-based CM systems demonstrated especially great performance compared to frequency-magnitude-based CM systems. In ASVspoof2021 [6], the issue of channel robustness was investigated by introducing channel variation on the evaluation set of ASVspoof2019. The results show strong performance degradation when evaluating on unseen channels. There have been some solutions provided by the participants of ASVspoof2021 [7, 8, 9, 10] that mainly use empirical methods to mitigate the problem, such as data augmentation and feature engineering.

In this paper, we propose to alleviate the channel robustness issue of state-of-the-art CM systems through phase perturbation in training. This proposal is developed based on two observations and our hypotheses for explaining the observations: 1) State-of-the-art CM systems are time-domain systems; We hypothesize that they rely on phase information to detect synthetic spoofing attacks, and 2) Communication channels often employ lossy compression codecs that are designed to only encode magnitude information; We hypothesize that they alter phase information in speech, making phase-aware CM systems difficult to generalize to unseen channels. Based on these hypotheses, we propose to perturb the phase when training phase-aware CM systems. We believe that this will make such systems less reliant on phase information, hence more robust to channel variations.

We design a set of experiments to test our hypotheses. By training three state-of-the-art time-domain CM systems and perturbing phase during evaluation, we discover that all CM systems' performance degrades as the phase perturbation amount increases. We also perturb magnitude during evaluation for comparison, and observe that better performing CM systems show a stronger degradation effect when the phase is perturbed compared to magnitude, suggesting that the performance boost in time-domain CM system is likely due to better utilization of phase information, yet such utilization becomes overfitting when phase is perturbed during evaluation.

Then, we employ the best performing CM system AASIST [11] and perturb phase during training, and evaluate on data from unseen transmission channels. We observe a significant performance improvement, where the best configuration demonstrates a 26.2% relative improvement on the equal error rate (EER). As a comparison, magnitude perturbation during training degrades performance in all settings. This suggests that the lossy compression in channel effects indeed retains magnitude information but corrupts phase information, and the phase perturbation in training is an effective strategy toward mitigating the channel robustness issues of time-domain CM systems. To our best knowledge, this is the first study that investigates the effects of phase and magnitude perturbation on CM systems. All code, audio samples, and trained model are open-sourced[1].

2. CM systems rely on phase information

In this section, we study the dependence of raw-waveform-based CM systems on phase information. We select three state-of-the-art CM systems with top performance on the ASVSpoof2019LA dataset [5]: RawNet-2 [12], RawGAT-ST [13], and AASIST [11].
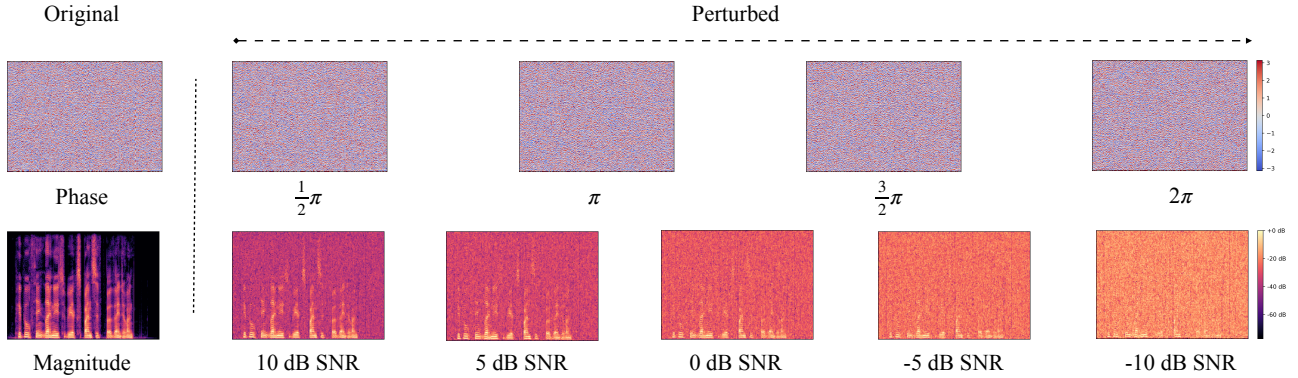
---

[1] https://yongyi.dev/phase-antispoofing

Figure 1: *Magnitude and phase spectrograms with different amounts of perturbation, using utterance LA_T_1000137 as an example. The horizontal axis is time from 0 to 2.5 s, while the vertical axis is frequency from 0 to 8 kHz.*

2.1. Dataset

We employ ASVspoof2019LA [14], which is the logical access (LA) subset of the ASVspoof2019 challenge. It contains bonafide speech from the VCTK corpus [15], with a vast variety of attacks. We follow the same training, development, and evaluation split as the ASVspoof2019 protocol. For the spoofing attacks, the training and development sets contain the same 6 attacks, and the evaluation set contains 13 unseen attacks.

2.2. Experimental setup

Phase perturbation. During evaluation, we randomly perturb the phase at different amounts. For an utterance with perturbation amount n, we extract its phase and magnitude spectrograms. The phase ϕ of every bin in the phase spectrogram is randomly reassigned a value in the range of $[\phi - \frac{n}{2}, \phi + \frac{n}{2}]$, while the magnitude spectrogram is unchanged. The phase perturbed utterance is then synthesized using the original magnitude and the perturbed phase spectrograms. We select four perturbation settings, with phase perturbation n as $\frac{1}{2}\pi$, π, $\frac{3}{2}\pi$ and 2π, respectively.

Magnitude perturbation. As a comparison, we also conduct a set of experiments with magnitude perturbed. For an utterance with a perturbation amount m dB signal-to-noise ratio (SNR), we add white noise to the utterance to make its SNR equal to m dB, then extract its magnitude spectrogram. The magnitude-perturbed utterance is then synthesized from the noisy magnitude spectrogram and the original utterance's phase spectrogram. If the synthesized audio waveform exceeds the maximum limit, we apply normalization to prevent clipping. We selected five perturbation settings for magnitude: 10 dB, 5 dB, 0 dB, -5 dB and -10 dB SNR. Figure 2 visualizes both phase and magnitude perturbation processes, while Figure 1 shows an example utterance with all perturbation settings. A setting without perturbation is also introduced as the baseline.

Evaluation metric. We follow the ASVspoof challenge series and employ equal error rate (EER), defined as the point where the false acceptance rate equals the false rejection rate. Lower EER indicates better performance.

Training details. All time-domain CMs are trained with the ASVspoof2019LA training set, and validated on the development set. To reproduce the best performance for all CM systems, we use the hyper-parameter settings as reported in [11], and select the checkpoint with the lowest validation EER within 100 epochs for evaluation. All models are trained with three dis-
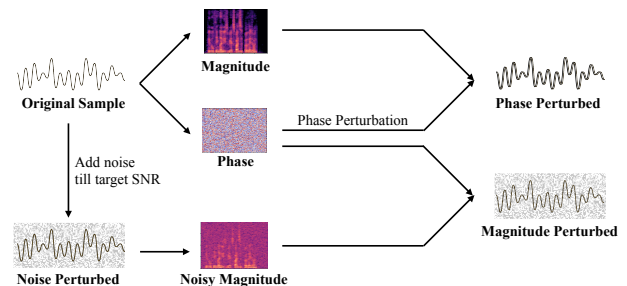

Figure 2: *Perturbation process for phase and magnitude corrupted data. For noise perturbation, the noise is directly added to the waveform.*

Table 1: *EER performance of three CM systems evaluated on phase perturbed test data.*

EER (%)		CM Systems		
		RawNet-2	RawGAT-ST	AASIST
No Perturbation		4.25	1.36	1.35
Phase	$\frac{1}{2}\pi$	9.55	20.09	21.12
	π	19.56	29.01	48.87
	$\frac{3}{2}\pi$	27.02	31.57	56.58
	2π	28.02	31.82	56.91
	Pooled	21.04	28.12	45.87

tinct random seeds and reported with an averaged result to mitigate the impact of the random seed on model performance [16].

2.3. Results and analyses

Table 1 demonstrates the EER performance for all three CM systems evaluated with phase perturbation. As a sanity check, the performance of all CM systems on ASVspoof2019LA is similar to that reported in [13] and [11]. As the amount of phase perturbation increases, the performance decreases on all three CM systems, indicating that time-domain CM systems utilize phase information to discern spoofed speech. As a comparison, Table 2 shows the results of all CM systems evaluated with different magnitude perturbation settings.

We use the pooled EER from all phase or magnitude per-

Table 2: *EER performance of three CM systems evaluated with magnitude perturbed test data.*

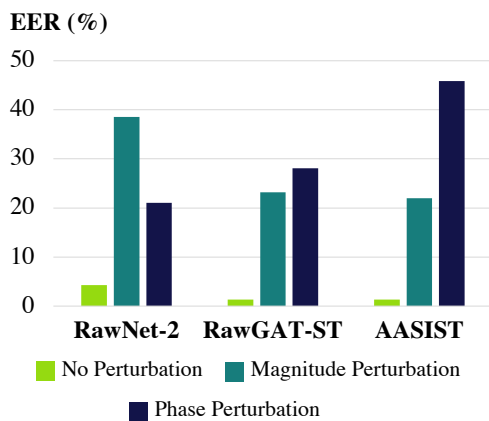| EER (%)		CM Systems	
	RawNet-2	RawGAT-ST	AASIST
No Perturbation	4.25	1.36	1.35
Mag. 10 dB	24.78	9.74	11.21
5 dB	34.08	15.11	13.01
0 dB	43.04	21.19	19.9
-5 dB	45.25	32.47	30.5
-10 dB	45.48	37.27	35.16
Pooled	38.53	23.16	21.96

Figure 3: *EER degradation in phase-perturbed and magnitude-perturbed settings compared to the baseline performance.*

turbed settings to represent the performance under this perturbation setting and compare them against the baseline setting. As illustrated in Figure 3, CM systems with better performance in ASVspoof2019LA show more degradation in phase-perturbed settings and less degradation in magnitude-perturbed settings, indicating that better-performing time-domain CM systems also rely more on phase information.

3. Phase perturbation during training improves channel robustness

In Section 2, our experiments establish the problem of overfitting to phase information in the trainin data.. In this section, we will explore perturbing phase during training to lessen phase reliance for time-domain CM to improve channel robustness.

Communication channels typically employ lossy compression codecs, many of which focus on encoding only frequency-magnitude information, since humans are more sensitive towards them [17]. After transmitting through such compression, much phase information is corrupted, making the performance of time-domain CM systems degrade as they have much reliance on phase information. By perturbing the phase during training, we can reduce the CM systems' reliance on phase information to build more robust CM systems. However, we expect that this perturbation should not be too much that completely removes the reliance on phase, as some useful phase information may still remain for the time-domain CM systems to pick up, even after going through the communication channel.

We hypothesize that there is a midway setting between not perturbing and perturbing all phases that provides the best trade-off resulting in best performance.

3.1. Channel-shifted dataset

To evaluate the performance of CM systems in unseen communication channels, we use ASVspoof2021LA, the logical access (LA) sub-track of the ASVspoof2021 challenge. This subset transmitted the entire ASVspoof2019LA set, along with additional samples, through seven communication channels, denoted as C1 to C7. C1 is the same channel as ASVspoof2019LA, while C2 to C7 are unseen channels. Amongst the unseen channels, C2 and C5 use the time-domain compression algorithms a-law and μ-law, while C4, C6, and C7 employ magnitude-based compression codecs, G.722 [18], GSM [19], and OPUS [20]. C3 differs from C2 by transmitting over a public switched telephone network, therefore introducing uncontrollable and unknown artifacts, such as data corruption during transit.

3.2. Experimental setup

As described in Section 2, all three CM systems heavily utilize phase information. In the interest of saving space while not losing generalization ability, we selected AASIST to study in depth due to its superior performance on ASVspoof2019LA.

We perturb the phase of the training portion of ASVspoof2019LA with the same four perturbation amounts: $\frac{1}{2}\pi$, π, $\frac{3}{2}\pi$ and 2π, then evaluate on ASVspoof2021LA. For magnitude, we use the same five magnitude perturbation settings as in Section 2. A baseline setting with no perturbation is also provided. Each setting is trained with three distinct random seeds and results are averaged. To better utilize GPU resources, the batch size of the experiments in this part is slightly increased, while all other hyperparameters remain unchanged.

Following the ASVspoof2021 challenge, when reporting performance on channel-variant data, we use pooled EER of all channels to represent the overall performance of CM systems.

3.3. Results and analysis

We begin by confirming our first hypothesis: by perturbing phase or magnitude during training, we can reduce the time-domain CM system's reliance on phase or magnitude. To do so, we train the CM system with different levels of phase or magnitude perturbation, and test it on phase or magnitude perturbed version of the ASVspoof2019LA evaluation set. For phase perturbed test data, we use the maximum phase perturbation (i.e., 2π), and for magnitude perturbed test data, we use -10 dB SNR for the perturbation. Since the perturbations are so strong in the test data, we expect to see performance degradation when there is no perturbation during training, then performance improvement as more phase perturbation is applied.

Results are as shown in Figure 4. In Figure 4(a), the performance improves with more phase perturbation in training, indicating less dependence on phase information. Similarly, Figure 4(b) illustrates the results on magnitude perturbation. We observe performance improvement as magnitude perturbation increases during training, indicating that the CM system is less dependent on magnitude information.

With this established, we perform evaluation of all training settings on ASVspoof2021LA, and the results are shown in Table 3. As expected, as magnitude is perturbed during training, performance on all unseen communication channels degrades,

Table 3: *Evaluation results on ASVspoof2021LA. The left side denotes the perturbation setting in training. C1-C7 denote different transmission codecs.*

| EER (%)		C1	C2	C3	C4	C5	C6	C7	Pooled
No Perturbation		4.68	5.87	14.39	5.75	5.44	7.66	10.26	9.91
Phase	$\frac{1}{2}\pi$	4.49	6.18	8.68	5.18	5.80	6.35	8.20	7.33
	π	6.72	7.00	7.34	6.41	6.89	6.85	7.30	7.31
	$\frac{3}{2}\pi$	5.52	6.20	10.66	5.21	6.18	7.25	6.21	8.32
	2π	5.68	6.37	9.63	5.42	6.34	7.14	6.39	7.73
Magnitude	10 dB	7.36	8.57	19.88	8.79	8.39	9.53	14.36	14.54
	5 dB	10.40	11.46	30.87	11.96	11.35	14.45	19.09	17.80
	0 dB	17.41	18.23	40.99	18.07	17.98	20.63	26.21	23.64
	-5 dB	23.70	25.05	46.63	24.45	25.00	29.75	34.23	29.87
	-10 dB	34.77	34.55	46.84	34.49	34.95	36.55	38.84	37.40

(a)   (b)
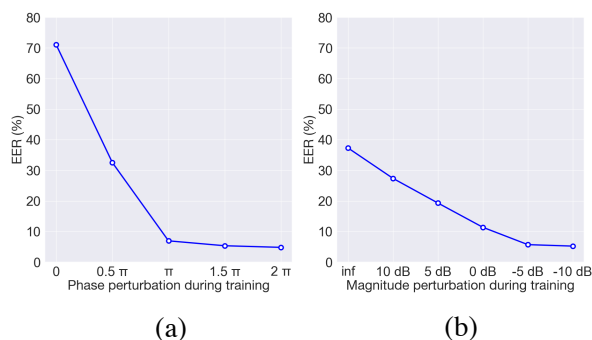
Figure 4: *(a) EER of AASIST trained with different levels of phase perturbation and evaluated on data with maximum phase perturbation (2π). (b) EER of AASIST trained with different levels of magnitude perturbation and evaluated on data with -10 dB SNR magnitude perturbation.*

showing that removing reliance on magnitude information is harmful for CM systems' performance. This suggests that CM systems can benefit from the magnitude information preserved by lossy compression codecs.

At the same time, all settings with phase perturbed show some improvement in pooled results, achieving a more robust overall performance compared to non-perturbed settings. This indicates that by being less sensitive to phase information, time-domain CM systems can better generalize to unseen communication channels. Best performing setting shows a relative EER improvement of 26.2% compared to the no perturbation baseline, without introducing any channel data during training. We also notice that the best-performing phase perturbation setting appears at π, aligning with to our hypothesis of a "midway" perturbation setting.

Amongst the individual unseen communication channels, C2 and C5 are codecs encoding the time-domain waveform directly, and we observe slight performance degradation as phase information is perturbed. C4, C6, and C7, on the other hand, uses encode magnitude spectrogram with lossy compression, and we see performance improvement with phase perturbation. Interestingly, we notice that on different channels, the best-performing phase perturbation setting appears at different per-

turbation amounts. This suggests that different amount of useful phase information is retained by different compression codecs.

C3 is transmitted through real-world phone lines and has more unknown data corruption en route, and we observe that phase perturbation during training improves performance. At different phase perturbation settings, C3 also shows more fluctuation, indicating that transmission artifacts are more likely to introduce phase artifacts as well.

On the C1 condition, even though no unseen communication artifacts are present, we still see a slight performance improvement at $\frac{1}{2}\pi$. We believe that this is likely because phase perturbation also brings better generalization ability to unseen attacks by masking part of model-specific artifacts. Speech generative systems typically take magnitude features as input and synthesize time-domain data; therefore, they have a heavier burden generating phase information. This makes speech generative algorithms more prone to creating phase artifacts that are model-specific. As we have established time-domain CM systems' reliance on phase information, we believe CM systems may have overfitted to specific artifacts in training data, and phase perturbation can also aid in mitigating this effect.

4. Conclusions

In this paper, we observed significant degradation of various state-of-the-art time-domain CM systems when evaluated on phase-perturbed speech utterances. This degradation may cause channel robustness issues, since many communication channels employ lossy codecs that only encode frequency-magnitude information while losing much phase information. We proposed to mitigate this issue by perturbing phase in training. Systematic evaluation on real-world channel-variant data verified that perturbing phase in training does significantly improve the channel robustness of a state-of-the-art time-domain CM system. For future work, we plan to use this insight to design CM systems that strike the balance between modeling useful phase information and being less sensitive to channel phase alternation.

5. Acknowledgement

6. References

[1] B. Sisman, J. Yamagishi, S. King, and H. Li, "An overview of voice conversion and its challenges: From statistical modeling to deep learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 132–157, 2021.

[2] X. Tan, T. Qin, F. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *arXiv preprint arXiv:2106.15561*, 2021.

[3] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li *et al.*, "Neural codec language models are zero-shot text to speech synthesizers," *arXiv preprint arXiv:2301.02111*, 2023.

[4] Y. Zhang, F. Jiang, G. Zhu, X. Chen, and Z. Duan, *Generalizing Voice Presentation Attack Detection to Unseen Synthetic Attacks and Channel Variation*. Springer, 2023, pp. 421–443. [Online]. Available: https://doi.org/10.1007/978-981-19-5288-3_15

[5] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, and H. Delgado, "ASVspoof 2021: accelerating progress in spoofed and deepfake speech detection," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 47–54.

[6] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgado, T. Kinnunen, M. Todisco, J. Yamagishi, N. Evans, A. Nautsch *et al.*, "ASVspoof 2021: Towards spoofed and deepfake speech detection in the wild," *arXiv preprint arXiv:2210.02437*, 2022.

[7] Z. Benhafid, S. A. Selouani, M. S. Yakoub, and A. Amrouche, "LARIHS ASSERT reassessment for logical access ASVspoof 2021 challenge," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 94–99.

[8] X. Chen, Y. Zhang, G. Zhu, and Z. Duan, "UR channel-robust synthetic speech detection system for ASVspoof 2021," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 75–82.

[9] T. Chen, E. Khoury, K. Phatak, and G. Sivaraman, "Pindrop labs' submission to the ASVspoof 2021 challenge," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 89–93.

[10] A. Tomilov, A. Svishchev, M. Volkova, A. Chirkovskiy, A. Kondratev, and G. Lavrentyeva, "STC antispoofing systems for the asvspoof2021 challenge," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 61–67.

[11] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, and N. Evans, "AASIST: Audio anti-spoofing using integrated spectro-temporal graph attention networks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6367–6371.

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

[13] H. Tak, J. weon Jung, J. Patino, M. Kamble, M. Todisco, and N. Evans, "End-to-end spectro-temporal graph attention networks for speaker verification anti-spoofing and speech deepfake detection," in *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, 2021, pp. 1–8.

[14] X. Wang, J. Yamagishi, M. Todisco, H. Delgado, A. Nautsch, N. Evans, M. Sahidullah, V. Vestman, T. Kinnunen, K. A. Lee, L. Juvela, P. Alku, Y.-H. Peng, H.-T. Hwang, Y. Tsao, H.-M. Wang, S. L. Maguer, M. Becker, F. Henderson, R. Clark, Y. Zhang, Q. Wang, Y. Jia, K. Onuma, K. Mushika, T. Kaneda, Y. Jiang, L.-J. Liu, Y.-C. Wu, W.-C. Huang, T. Toda, K. Tanaka, H. Kameoka, I. Steiner, D. Matrouf, J.-F. Bonastre, A. Govender, S. Ronanki, J.-X. Zhang, and Z.-H. Ling, "ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech," *Computer Speech & Language*, vol. 64, pp. 10,11–14, 2020.

[15] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2017.

[16] X. Wang and J. Yamagishi, "A comparative study on recent neural spoofing countermeasures for synthetic speech detection," in *Proc. Interspeech*, 2021, pp. 4259–4263.

[17] T. Painter and A. Spanias, "Perceptual coding of digital audio," vol. 88, no. 4, pp. 451–515.

[18] "G.722 : 7 khz audio-coding within 64 kbit/s," 1988, accessed on Feb 28th, 2023. [Online]. Available: https://www.itu.int/rec/T-REC-G.722/_page.print

[19] K. Jarvinen, J. Vainio, P. Kapanen, T. Honkanen, P. Haavisto, R. Salami, C. Laflamme, and J.-P. Adoul, "GSM enhanced full rate speech codec," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, 1997, pp. 771–774.

[20] J.-M. Valin, K. Vos, and T. Terriberry, "Definition of the opus audio codec," Tech. Rep., 2012.