



Defend for Self-Vocoding: A Novel Enhanced Decoder Network for Watermark Recovery

Yu-Sheng Lin, Ching-Yu Yang, Hsing-Hang Chou, Ya-Tse Wu, Bo-Hao Su, Chi-Chun Lee

Department of Electrical Engineering, National Tsing Hua University, Taiwan

yshlin1108@gapp.nthu.edu.tw, gino0950150@gmail.com, stargazer@gapp.nthu.edu.tw,
crowpeter@gapp.nthu.edu.tw, borrisu@gapp.nthu.edu.tw, ccleee@ee.nthu.edu.tw

Abstract

Recent advances in voice cloning technology have raised security concerns due to its ability to generate highly realistic synthetic speech, making it challenging to detect malicious usage. Proactive watermarking approaches embed authentication information in target voices to prevent unauthorized synthesis. While existing methods show resilience against traditional preprocessing attacks, we identify a novel threat, *self-vocoding*, which reconstructs audio using neural vocoders, can cause severe watermark degradation but preserve high audio fidelity. To address this, we propose an enhanced decoding framework to handle self-vocoding distortions on watermarks. In addition to general vocoder distortions, we systematically categorize them into two vocoder types for further analysis. Experimental results demonstrate that our approach significantly improves watermark decoding accuracy, offering an effective defense against self-vocoding attacks.

Index Terms: voice cloning, watermark recovery, vocoder, self-vocoding

1. Introduction

Audio deepfake technology, or voice cloning, enables the generation of synthetic speech that closely mimics specific individuals, including phrases they never spoke. Initially, this effort was developed with intentions to positively advance speech interface technologies. However, the rapid advancements in text-to-speech (TTS) and voice conversion (VC) [1, 2, 3, 4] have introduced unwanted security risks. These models can synthesize human voices with such high fidelity that they are often indistinguishable from real ones, raising concerns about misuse. For example, in 2020, scammers used voice cloning to impersonate a CEO, successfully deceiving a branch manager into transferring \$35 million [5], highlighting the urgent need for preventing unauthorized voice synthesis. Efforts to address these challenges fall into two categories; passive detection methods [6, 7, 8] aim to identify spoof voices but risk creating an endless race between synthesis and detection technologies. Alternatively, proactive defense strategies have been proposed. For instance, Huang *et al.* [9] and Yang *et al.* [10] added adversarial noises to target voices, causing synthetic outputs to deviate significantly in timbre. However, these methods often require prior knowledge of the attacker’s cloning model, limiting their generalizability, and may distort the protected audio’s quality.

To address these limitations, recent works have explored proactive methods of embedding imperceptible watermarks (e.g., indicating ownership or authentication) into target voices before public release. However, audio processing is pervasive—whether intentional or unintentional, any transformation or transmission of audio files may distort the embedded water-

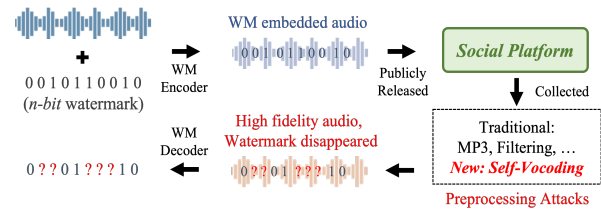


Figure 1: “Self-vocoding” watermark preprocessing attack.

mark. Malicious actors can easily exploit this vulnerability by applying *watermark preprocessing attacks* to distort the watermark. This situation has recently caught attentions by the researchers, e.g., Roman *et al.* [11] and Liu *et al.* [12] utilize deep neural networks that demonstrate improved robustness in watermark recovery against various audio preprocessing attacks. The preprocessing techniques they examined include traditional audio processing such as re-sampling, lossy compression, and frequency filtering. Among these, high-pass filtering (Roman *et al.* [11]) and MP3 compression and low-pass filtering (Liu *et al.* [12]) have been identified as particularly harmful to watermark integrity due to their significant impact on audio quality.

However, vocoders, which have already become the de-facto components for converting spectrograms into high quality waveforms for a wide range of speech generation technologies, have posed a significant threat by our investigation. We show that the method of *self-vocoding*, which leverages neural vocoders to reconstruct original audio with high fidelity while preserving speaker similarity, is surprisingly more harmful to the embedded watermarks than those previously examined preprocessing attacks. To the best of our knowledge, this is the first work to point out that neural vocoding can act as a powerful watermark distortion tool (Figure 1). In this work, we further propose an enhanced watermark decoding network to prevent self-vocoding attacks on watermarks.

Building upon the Timbre Watermarking embedding architecture proposed by Liu *et al.* [12], we further introduce an additional reconstruction model that handles unwanted distortions to maintain high-fidelity watermark recovery. The use of an enhanced reconstruction model is inspired by image-to-image reconstruction methods while exposed to unknown adversarial attacks [13]. In our work, we integrate a similar reconstruction model into our decoding pipeline to recover watermark-informative spectrograms before feeding into watermark decoder. We examined one general and two specialized reconstruction models each trained on data specific to either autoregressive or non-autoregressive vocoder distortions. Experimental results show that our framework significantly improves watermark decoding accuracy against critical self-vocoding attacks, offering an effective solution to this emerging threat.

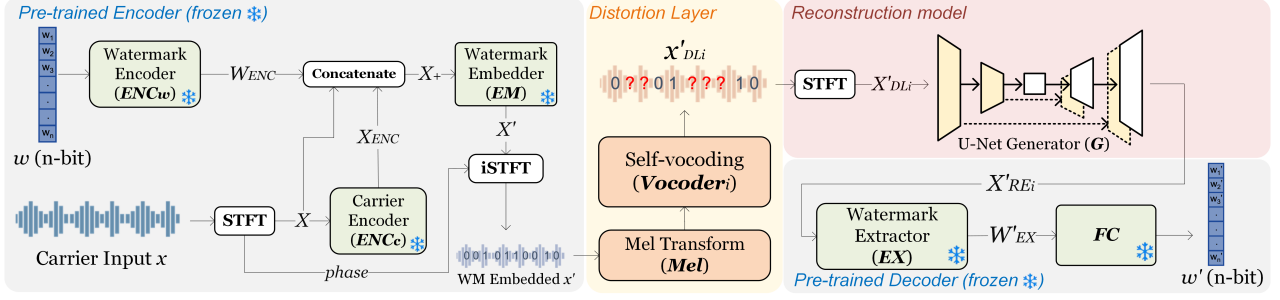


Figure 2: A watermark embedding encoder-distortion-decoder network. A modified distortion layer is introduced with “self-vocoding” of different vocoders, and a reconstruction model is integrated to recover watermark-informative spectrograms before decoding.

2. Methodology

The proposed framework is illustrated in Figure 2. Our objective is to accurately decode the n -bit watermark when a watermarked audio has experienced *self-vocoding* distortion. We build upon the Timbre Watermarking architecture [12], which adopts an *Encoder-Distortion Layer-Decoder* structure, with modifications to the distortion layer and the decoding pipeline. Specifically, we introduce a U-Net generator G before the original decoder network to learn the mapping from vocoder-distorted watermarked spectrograms to their original watermarked counterparts. The watermark encoder and decoder are pre-trained models¹ from [12], and only the newly plugged-in watermark reconstruction model needs to be trained.

2.1. Pre-trained Encoder

The encoder aims to embed a speech signal with imperceptible watermark information. The widely used linear spectrogram [14] of speech audio is adopted as the carrier for robust watermark embedding. Given a single-channel raw speech audio signal x , Short-Time Fourier Transform (STFT) is used to obtain its spectrogram X and phase. The magnitude spectrogram X is then passed as an input carrier to the Carrier Encoder module ENC_c , generating the encoded carrier feature map X_{ENC} . Meanwhile, the n -bit watermark w is fed into the Watermark Encoder module ENC_w , producing the encoded watermark features, then repeated along the time domain to match the size of X_{ENC} , resulting in W_{ENC} :

$$W_{ENC} = \text{Repeat}(ENC_w(w), T) \quad (1)$$

Inspired by DenseNet [15], the original spectrogram X is concatenated with X_{ENC} and W_{ENC} to preserve the original carrier information to the greatest extent:

$$X_+ = \text{Concatenate}(W_{ENC}, X, X_{ENC}) \quad (2)$$

The combined feature map X_+ is then passed through the Watermark Embedder module EM , producing the watermark-embedded spectrogram X' . Finally, the watermark-embedded speech audio x' is reconstructed by applying the Inverse Short-Time Fourier Transform (iSTFT) on X' and the original phase.

2.2. Distortion Layer

To handle the newly identified self-vocoding attack, our pipeline incorporates a modified distortion layer to simulate different kinds of vocoder distortions:

$$x'_{DL_i} = \text{Vocoder}_i(\text{Mel}(x')) \quad (3)$$

where Mel and i denote the mel-transform and each type of neural vocoders used in our experiment, respectively.

¹<https://github.com/TimbreWatermarking/TimbreWatermarking>

2.3. Reconstruction Model (U-Net generator)

We integrate a reconstruction model into our decoding pipeline to restore watermark-informative spectrograms before feeding into the decoder. This model is inspired by Pix2pix [16], a well-known model for image translation tasks. It consists of a U-Net-like generator, an encoder-decoder network with skip connections between mirrored layers in the encoder and decoder, and a discriminator to differentiate between real and generated images. It has demonstrated its powerful ability to map a high-resolution input grid to a high-resolution output grid with the same underlying structure in both the input and output images. Given that self-vocoded audio exhibits a similar underlying spectrogram structure, we adopt this model to handle our task of reconstructing watermark-informative spectrograms. Specifically, we modify the generator size to match our spectrogram size and use it to reconstruct the distorted spectrogram. The mapped spectrogram output can be formulated as:

$$X'_{RE_i} = G(X'_{DL_i}, z) \quad (4)$$

where X'_{DL_i} is the magnitude spectrogram of x'_{DL_i} and G denotes the U-Net generator taking the distorted spectrogram as input while conditioning on the dropout noise z with a 50% dropout rate in the Convolution-BatchNorm-Dropout-ReLU layers. Note that our goal is to recover the watermark-informative spectrogram, not to focus on distinguishing between real or fake mapped spectrograms. Therefore, we omit the discriminator; otherwise, the generator could introduce unnecessary noise into the reconstructed spectrogram. Hence, the $L1$ loss (or pixel loss) used to constrain this model can be written as:

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{X', X'_{DL_i}, z} [\|X' - G(X'_{DL_i}, z)\|_1] \quad (5)$$

To train this model, the pre-trained encoder is first used to embed random watermarks into speech samples from the entire training set. These watermark-embedded samples are then processed through the distortion layer, where each vocoder generates self-vocoded audio, resulting in paired original watermark-embedded and watermark-distorted spectrograms. The reconstruction model is subsequently trained to learn the mapping between these paired samples using the $L1$ loss.

2.4. Pre-trained Decoder

The decoder is then employed to extract the final watermark bits. Once the watermark-informative spectrogram X'_{RE_i} is reconstructed, it is processed by the Watermark Extractor module EX to obtain the watermark embedding W'_{EX} , which is then averaged along the time axis and fed into the final fully connected layer FC to decode the watermark bits:

$$w' = FC(\text{Average}(W'_{EX})) \quad (6)$$

Note that this average operation corresponds to the previous repeat operation in Eq. 1 and if the value of w' is greater than or equal to 0, the corresponding bit is decoded as 1; if w' is less than 0, the bit is decoded as 0.

3. Experiment Setup

3.1. Dataset

We employ LJSpeech², a single speaker dataset with a 22.05kHz audio sampling rate, as the dataset in our experiment. The dataset contains 13100 audio clips with length varies from 1 to 10 seconds and a total of nearly 24 hours. We further split the dataset into training and testing sets in a ratio of 9:1.

3.2. Distortion Types and Training Setup

Vocoders can be categorized into two main types: autoregressive and non-autoregressive. Autoregressive vocoders predict each audio sample based on previous ones, producing high-quality speech. In contrast, non-autoregressive vocoders generate samples in parallel for higher efficiency but with lower synthesis quality compared to autoregressive models. For this study, we consider three vocoders from each category: WaveRNN [17], WaveFlow [18], and CARGAN [19] for autoregressive models, and HiFiGAN [20], DiffWave [21], and iSTFTNet [22] for non-autoregressive models.

Since there is a fundamental difference between autoregressive and non-autoregressive vocoders, we train three versions of the reconstruction model. For the autoregressive version, we use WaveRNN and WaveFlow as *seen* distortions; for the non-autoregressive version, HiFiGAN and DiffWave serve as *seen* distortions. Meanwhile, in the general (mixed) version, all four vocoders are used as *seen* distortions. CARGAN and iSTFTNet are retained as *unseen* distortions for all versions to evaluate the models' robustness across different vocoder types. For a baseline comparison, instead of plugging in our proposed reconstruction model, we simply finetune three versions of the original decoder from [12] on the same setting using its original watermark decoding loss.

3.3. Evaluation Metrics

To evaluate audio fidelity when comparing traditional harmful preprocessing methods with the *self-vocoding* attacks, we employ two objective evaluation metrics. First, we use **DNSMOS** [23], a non-intrusive perceptual objective speech quality metric designed to evaluate noise suppressors, which ranges from 1 to 5, as our measure of perceptual speech quality. Second, for speaker similarity, we utilize Resemblyzer [24] to extract 256-dimensional speaker embeddings and calculate speaker embedding cosine similarity (**SECS**).

For assessing the effectiveness of watermark extraction, we measure the bit recovery accuracy (**ACC**). All metrics are computed on the entire test set, consisting of 1,300 audio files. For **ACC** presented in all tables, each speech sample is embedded with 2 random watermarks as in [12], resulting in a total of 2,600 speech samples. All evaluations are repeated three times and a standard deviation is reported.

3.4. Implementation Details

The code and audio samples can be found on our github repository³. The default watermark length n is set to 10 bits. For **ENC_w**, **EM**, and **EX**, we use the approach described in [12], utilizing fully 2D convolutional networks that combine

²<https://keithito.com/LJ-Speech-Dataset/>

³<https://github.com/dabaobi/Self-Vocoding-Defender>

Table 1: *Audio fidelity comparison between traditional harmful preprocessing versus self-vocoding.*

Distortion	ACC (%) ↓	DNSMOS ↑	SECS ↑
MP3	94.97 (±1.05)	3.087 (±0.006)	0.806 (±0.004)
Filtering	95.11 (±0.76)	3.388 (±0.004)	0.748 (±0.005)
WaveRNN	86.39 (±1.79)	4.015 (±0.004)	0.988 (±0.001)
WaveFlow	94.48 (±1.18)	3.982 (±0.007)	0.994 (±0.000)
CARGAN	72.79 (±1.58)	3.970 (±0.004)	0.983 (±0.002)
HiFiGAN	67.75 (±1.97)	3.969 (±0.012)	0.946 (±0.002)
DiffWave	94.27 (±1.62)	3.851 (±0.004)	0.971 (±0.001)
iSTFTNet	75.99 (±1.21)	3.794 (±0.006)	0.942 (±0.004)

Gated Convolutional Neural Networks [25] with skip connections [26]. The watermark encoder, **ENC_w**, is implemented as a fully connected layer with a LeakyReLU activation function. For the Short-Time Fourier Transform (STFT), we use a filter length of 1024, a hop length of 256, and a window of length 1024.

In the U-Net generator **G**, all convolutions are 4×4 spatial filters applied with stride 2, and convolutions in the encoder downsample by a factor of 2, whereas in the decoder they up-sample by a factor of 2. The number of downsampling layers is increased to 9 to fit spectrogram inputs. The dropout noise z is not applied during inference. For training details, we use a batch size of 4 and the Adam optimizer [27] with a learning rate of 0.0002 and momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The model is trained for 200 epochs on a Nvidia GeForce RTX 3090 GPU within 72 hours. Model parameters are initialized using the default settings in PyTorch 1.13.0 [28].

4. Results and Analysis

In the following paragraph, we first present a comparison between traditional harmful preprocessing attacks and the novel self-vocoding preprocessing attack in our experiment. Secondly, we discuss the performance on the watermark decoding accuracy. Note that for other traditional preprocessing methods, Timbre Watermarking [12] has shown a strong ability to achieve nearly 100% decoding accuracy, thus they are not included in further consideration.

4.1. Harmful Preprocessing v.s. Self-vocoding

The comparison results are shown in Table 1. As highlighted in [12], MP3 8kbps compression and 2000Hz low-pass filtering are identified as the most harmful traditional preprocessing methods. Despite severely degrading audio quality, these techniques still retain approximately 95% decoding accuracy (ACC) for watermark extraction. In contrast, for all vocoders in our experiment, self-vocoding results in even greater watermark loss, however, maintains a much higher audio fidelity, measured by MOS and speaker similarity (SECS) scores. Notably, CARGAN, HiFiGAN, and iSTFTNet lead to the most dramatic ACC drop of nearly 30% or more, while achieving a MOS score of nearly 4 and preserving speaker similarity with SECS values very close to 1. Practically, for voice authentication, an SECS above 0.9 is sufficient to pass speaker verification, highlighting the significant potential risks posed by self-vocoding as a watermark preprocessing attack method.

4.2. Watermark Decoding Accuracy

4.2.1. Mixed Version

The results of the proposed enhanced decoding network are presented in Table 2. As shown in the results of the mixed version, our enhanced watermark decoding network consistently achieves higher decoding accuracy across all vocoder types

Table 2: Watermark decoding accuracy (ACC) comparison. The best performance in each column is marked in bold.

	WaveRNN	WaveFlow	HifiGAN	DiffWave	CARGAN	iSTFTNet	Average
Vocoder type	Auto.		Non-auto.		Auto.	Non-auto.	
Original ACC	86.39 (± 1.79)	94.48 (± 1.18)	67.75 (± 1.97)	94.27 (± 1.62)	72.79 (± 1.58)	75.99 (± 1.21)	81.95
Mixed ver.	Seen		Seen		Unseen	Unseen	
Finetune orig. decoder	92.96 (± 1.03)	96.55 (± 0.32)	85.70 (± 0.88)	97.32 (± 0.13)	74.28 (± 1.17)	83.96 (± 1.11)	88.46
Proposed	99.39 (± 0.26)	99.30 (± 0.05)	98.77 (± 0.33)	98.89 (± 0.07)	78.72 (± 1.92)	85.92 (± 1.83)	93.50
Auto. ver.	Seen		Unseen		Unseen	Unseen	
Finetune orig. decoder	93.25 (± 1.34)	96.77 (± 0.37)	77.29 (± 2.19)	93.89 (± 1.48)	76.15 (± 0.53)	82.29 (± 1.09)	86.61
Proposed	99.67 (± 0.04)	99.81 (± 0.02)	88.39 (± 1.06)	96.46 (± 1.26)	82.58 (± 1.61)	83.98 (± 1.77)	91.82
Non-auto. ver.	Unseen		Seen		Unseen	Unseen	
Finetune orig. decoder	91.54 (± 1.42)	94.86 (± 1.16)	87.38 (± 0.89)	97.55 (± 0.02)	73.16 (± 1.68)	86.89 (± 1.09)	88.56
Proposed	94.28 (± 1.39)	96.10 (± 0.18)	99.03 (± 0.15)	99.25 (± 0.06)	74.61 (± 1.56)	89.56 (± 1.27)	92.14

compared to simply finetuning the original decoder, demonstrating the necessity of the reconstruction model in effectively restoring watermark-informative spectrograms. Notably, for all **seen** vocoder types, the enhanced decoding network reaches almost 100% decoding accuracy. Specifically, in HiFiGAN, the ACC has a drastic improvement of 31.02%, highlighting the reliability of our proposed approach. For **unseen** vocoder types, our enhanced decoding network also achieves significant improvements. The ACC for CARGAN increases 5.93%, while for iSTFTNet, it improves 9.93%. These results show the effectiveness of our model to ‘clean-up’ vocoder distortions in the spectrogram, further generalizing to those unseen vocoders.

4.2.2. Autoregressive and Non-autoregressive Versions

We additionally train two specialized versions of the reconstruction model for comparison. As shown in Table 2, within the same column, the autoregressive version achieves the best performance on **seen** autoregressive vocoders (e.g., WaveRNN, WaveFlow), while the non-autoregressive version shows superior performance on **seen** non-autoregressive vocoders (e.g., HiFiGAN, iSTFTNet). Moreover, each model demonstrates better generalizability for the **unseen** vocoder within its respective category. For instance, in the same column, the autoregressive version achieves the highest decoding accuracy (82.58%) on CARGAN, while the non-autoregressive version performs best (89.56%) on iSTFTNet. Furthermore, we observed that for the two **unseen** vocoders, the autoregressive model achieves a total improvement of 17.78%, which is the largest increase among all three versions. An interesting point is that the autoregressive model seems to generalize better to non-autoregressive model—it achieves 83.98% ACC on unseen non-autoregressive iSTFTNet, while non-autoregressive model achieves only 74.61% ACC on unseen autoregressive CARGAN.

4.2.3. Watermark Embedding Visualization

We further visualize the extracted watermark embedding W'_{EX} , averaged across the entire test set with the same embedded watermark pattern, to analyze the distorted watermark introduced by *self-vocoding* of different vocoders and demonstrate the effectiveness of the reconstruction model. We focus on the two unseen vocoders, CARGAN and iSTFTNet, and use the reconstruction model trained on mixed vocoder types to perform watermark reconstruction. Figure 3(a) shows the extracted watermark embedding from the original watermarked audio as ground truth. Figures 3(b) and 3(d) present the extracted watermark embeddings after distorted by CARGAN and iSTFTNet, respectively. In contrast, Figures 3(c) and 3(e) show the results after applying the reconstruction model. As highlighted in

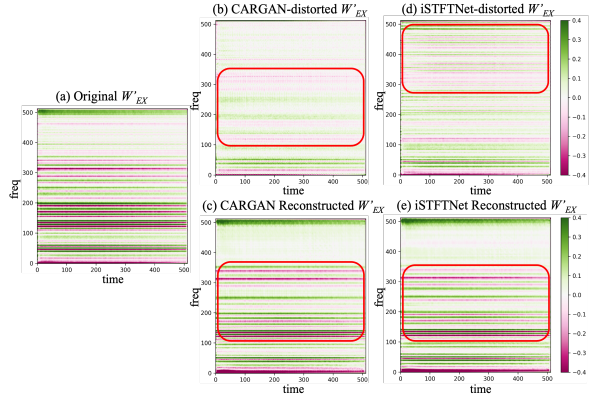


Figure 3: Extracted watermark embedding comparison. (c) is the reconstructed version of (b), while (e) corresponds to (d).

Figure 3(b), CARGAN, being an autoregressive vocoder, tends to erase watermark information primarily in the 100th to 300th frequency bin range, likely due to its focus on generating high-quality audio synthesis. Meanwhile, iSTFTNet not only disrupts the watermark in the same frequency range but also introduces undesired noise in higher frequency regions (above the 300th frequency bin), as marked in Figure 3(d), complicating watermark decoding. However, after passing through our reconstruction model (shown in Figures 3(c) and 3(e)), the watermark information is effectively restored, especially within the 100th to 350th frequency bin range. The restoration process further demonstrates that the watermark in the spectrogram is not entirely erased but rather distorted and spread across the spectrogram by the vocoders. This underscores the crucial role of the reconstruction model in recovering the embedded watermark and may explain why our model significantly improves decoding accuracy for these challenging vocoder distortions.

5. Conclusion

In this work, we show that the newly-identified watermark pre-processing attack using neural vocoders can significantly degrade watermark decoding accuracy but preserve high audio fidelity and speaker similarity. To counter this threat, we propose an enhanced watermark decoding network that incorporates a reconstruction model inspired by image-to-image translation models. Our pipeline consistently improves decoding accuracy, achieving nearly 100% on seen vocoder distortions and significantly enhancing performance on unseen vocoders. Future work includes exploring adaptive reconstruction models for better generalization and extending evaluation to multi-speaker datasets for broader applicability.

6. References

- [1] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/4559912e7a94a9c32b09d894f2bc3c82-Paper.pdf
- [2] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5530–5540.
- [3] Y. Yang, Y. Kartynnik, Y. Li, J. Tang, X. Li, G. Sung, and M. Grundmann, "Streamvc: Real-time low-latency voice conversion," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 11 016–11 020.
- [4] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar *et al.*, "Voicebox: Text-guided multilingual universal speech generation at scale," *Advances in neural information processing systems*, vol. 36, 2024.
- [5] T. Brewster, "Fraudsters cloned company director's voice in \$35 million bank heist, police find." *Forbes*. 2022.
- [6] M. E. Ahmed, I.-Y. Kwak, J. H. Huh, I. Kim, T. Oh, and H. Kim, "Void: A fast and light voice liveness detection system," in *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 2685–2702. [Online]. Available: <https://www.usenix.org/conference/usenixsecurity20/presentation/ahmed-muhammad>
- [7] A. Lieto, D. Moro, F. Devoti, C. Parera, V. Lipari, P. Bestagini, and S. Tubaro, "'hello? who am i talking to?" a shallow cnn approach for human vs. bot speech classification," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 2577–2581.
- [8] R. Wang, F. Juefei-Xu, Y. Huang, Q. Guo, X. Xie, L. Ma, and Y. Liu, "Deepsonar: Towards effective and robust detection of ai-synthesized fake voices.(2020)," in *Proceedings of the 28th ACM International Conference on Multimedia, MM*, 2020, pp. 12–16.
- [9] C.-y. Huang, Y. Y. Lin, H.-y. Lee, and L.-s. Lee, "Defending your voice: Adversarial attack on voice conversion," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 552–559.
- [10] C.-Y. Yang, S. G. Upadhyay, Y.-T. Wu, B.-H. Su, and C.-C. Lee, "Rw-voiceshield: Raw waveform-based adversarial attack on one-shot voice conversion," in *Interspeech 2024*, 2024, pp. 2730–2734.
- [11] R. San Roman, P. Fernandez, H. Elsahar, A. Défossez, T. Furon, and T. Tran, "Proactive detection of voice cloning with localized watermarking," in *International Conference on Machine Learning*, vol. 235, 2024.
- [12] C. Liu, J. Zhang, T. Zhang, X. Yang, W. Zhang, and N. Yu, "Detecting voice cloning attacks via timbre watermarking," 2023. [Online]. Available: <https://arxiv.org/abs/2312.03410>
- [13] H. Zhang, Z. Yao, and K. Sakurai, "Versatile defense against adversarial attacks on image recognition," 2024. [Online]. Available: <https://arxiv.org/abs/2403.08170>
- [14] F. Itakura, "Line spectrum representation of linear predictor coefficients of speech signals," *The Journal of the Acoustical Society of America*, vol. 57, no. S1, pp. S35–S35, 08 2005. [Online]. Available: <https://doi.org/10.1121/1.1995189>
- [15] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.
- [16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.
- [17] N. Kalchbrenner, E. Elsen, K. Simonyan, S. Noury, N. Casagrande, E. Lockhart, F. Stimberg, A. Oord, S. Dieleman, and K. Kavukcuoglu, "Efficient neural audio synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2410–2419.
- [18] W. Ping, K. Peng, K. Zhao, and Z. Song, "WaveFlow: A compact flow-based model for raw audio," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 7706–7716. [Online]. Available: <https://proceedings.mlr.press/v119/ping20a.html>
- [19] M. Morrison, R. Kumar, K. Kumar, P. Seetharaman, A. Courville, and Y. Bengio, "Chunked autoregressive gan for conditional waveform synthesis," 2022. [Online]. Available: <https://arxiv.org/abs/2110.10139>
- [20] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in neural information processing systems*, vol. 33, pp. 17 022–17 033, 2020.
- [21] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "Diffwave: A versatile diffusion model for audio synthesis," in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=a-xFK8Ymz5J>
- [22] T. Kaneko, K. Tanaka, H. Kameoka, and S. Seki, "istftnet: Fast and lightweight mel-spectrogram vocoder incorporating inverse short-time fourier transform," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 6207–6211.
- [23] C. K. Reddy, V. Gopal, and R. Cutler, "Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6493–6497.
- [24] C. Jemine, "resemble-ai-resemblelyzer," 2023. [Online]. Available: <https://github.com/resemble-ai/Resemblelyzer>
- [25] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML'17. JMLR.org, 2017, p. 933–941.
- [26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.
- [27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: <https://api.semanticscholar.org/CorpusID:6628106>
- [28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.