FIRM: DETECTING ADVERSARIAL AUDIOS BY RECURSIVE FILTERS WITH RANDOMIZATION

Guanhong Tao[§],[*] Xiaowei Chen[†], Yunhan Jia[†], Zhenyu Zhong[†], Shiqing Ma[‡], Xiangyu Zhang[§]
[§]Purdue University, [†]Baidu USA, [‡]Rutgers University

ABSTRACT

There is an emerging trend of exploring adversarial attacks against automatic speech recognition (ASR) systems. Existing defense techniques largely adopt methods in the image processing domain without considering the domain-specific characteristics of audio signals. Hence, they have various limitations against advanced attacks. In this paper, we study the characteristics of (adversarial) audio signals in the frequency domain and identify that existing techniques heavily rely on high frequency perturbation that is not perceptible by humans. We hence leverage recursive filters that are widely used in the signal processing to filter high frequency signal. To harden possible future attack on the low frequency, we further introduce randomness in our recursive filter such that it becomes extremely difficult to evade. Our experiments show that our technique FIRM can effectively detect adversarial attacks with AUC scores greater than 0.96 on five different attack scenarios and two ASR systems. It is also resilient to adaptive attacks. In contrast, a state-of-the-art approach can only achieve an AUC score of 0.58 on some advanced attacks.

1 INTRODUCTION

Recent deep neural networks (DNNs) based *automatic speech recognition* (ASR) systems, such as DeepSpeech (Hannun et al., 2014) and Lingvo (Shen et al., 2019), have achieved state-of-the-art performance on speech-to-text tasks. Just like image classification models are vulnerable to adversarial attacks, DNN-based ASRs are also subject to attacks by adversarial audios. Recently, Qin et al. (2019) constructed human-imperceptible adversarial audios that lead to a *target* transcription chosen by the attacker. This raises the security concerns regarding commercial ASR systems such as Google Home and Amazon Alexa whose core architectures are based on DNNs.

Most existing defense techniques utilize similar methods as those in the image processing domain without considering domain specific characteristics of audio signals. For instance, Zeng et al. (2019) leveraged the outputs of multiple ASRs to detect adversarial audios. It has been shown (in the image processing domain) that defense using multiple models is susceptible to adaptive attacks, which are aware of defense techniques (He et al., 2017). The techniques in (Rajaratnam & Kalita, 2018; Das et al., 2018) employed compression methods, e.g., MP3 compression. However, these techniques may not be effective (under adaptive attacks), as shown by Athalye et al. (2018). Yang et al. (2019) observed that an adversarial audio often contains substantial dependency within the entire signal such that breaking part of the dependency yields very different transcription. Hence, they proposed to remove the later half of a given audio signal and compare the resulted transcription of the remaining half to the first half of the transcription generated from the whole input signal. Substantial differences indicate attacks. Their results show that the technique is highly effective and represents the state-of-the-art. However, as shown later in Section 4, the assumption that adversarial signals have prevalent dependence may not hold in general.

In speech recognition, a raw audio signal is first transformed to frequency spectrum, which is then passed to the DNN of an ASR systems to generate transcription. That is, the actual input to ASR's DNN models is in the frequency domain. It is hence more productive to study the characteristics of audio signals in the frequency domain instead of in the raw time domain, in order to develop effective defense techniques. We observe that existing adversarial attacks rely on injecting high-frequency

[*] Work done as an intern at Baidu USA.

noise, which is not human perceptible. We hence propose to leverage the widely used recursive filters (RFs) that are designed for low-pass filtering. While existing attacks heavily rely on perturbation in the high frequency, we further show that adaptive attacks that are aware of the existence of recursive filters can succeed by perturbing the low frequency. To harden low frequency signals, we introduce randomness into RF, where the coefficients of RF are re-sampled from some distribution on the fly after every short time period. As such, the randomized RF processes the input signal in potentially infinite number of ways that are unknown beforehand. Our technique FIRM (recursive **FI**lters with **R**ando**M**ization) is comprised of two parallel processing paths: (1) the input audio waveform is directly fed to the ASR; (2) the same audio first goes through a randomized RF before being passed to the ASR. The two transcribed sentences are contrasted using the *word error rate* (WER). Input audio with a large WER is considered adversarial. In summary, we make the following contributions.

- We propose to detect adversarial audios utilizing their characteristics in the frequency domain.
- We incorporate randomness in a recursive filter for defending against potential adaptive attacks that are aware of our defense.
- Our technique FIRM can successfully detect four different adversarial attacks with AUC scores greater than 0.96, whereas a state-of-the-art approach (Yang et al., 2019) is only effective on some of the attacks and ineffective on advanced attacks with an AUC score of 0.58. In addition, FIRM is resilient to adaptive attacks with negligible performance degradation.

2 BACKGROUND

Frequency Domain of ASR. Unlike image recognition systems where image pixels are directly used for prediction, audio utterances are first transformed from the time domain to the frequency domain. The process involves dividing the input utterance into overlapping frames with a fixed time span (e.g., 25 ms each frame with 15 ms overlap). Each frame then goes through an N-point Fast Fourier-Transformation (FFT) to obtain the frequency spectrum. This is called the *Short-Time Fourier-Transformation* (STFT). Due to the non-linearity of human ear's perception of sound, *Mel-Frequency Cepstrum* (MFC) transformation is applied to make features more discriminative at lower frequencies and less discriminative at higher frequencies.

Recursive Filters. In signal processing, digital filters are commonly adopted for separating signals or recovering distorted signals (Smith et al., 1997). There are two categories of digital filters: *non-recursive* and *recursive*. A non-recursive filter computes output values exclusively from input values. A recursive filter, on the other hand, utilizes both input values and prior outputs in computation. Formally, suppose $\mathbf{x} = (x_0, x_1, ..., x_n)$ is the input samples and $\mathbf{y} = (y_0, y_1, ..., y_n)$ is the corresponding output values generated by a recursive filter at the corresponding sample points, then the recursive filter leverages the following equation to compute the current output y_i:

$$\begin{aligned} y_i = a_0 x_i + a_1 x_{i-1} + a_2 x_{i-2} + ... + a_i x_0 \\ + b_1 y_{i-1} + b_2 y_{i-2} + ... + b_i y_0, \end{aligned} \tag{1}$$

where $a_0, a_2, ..., a_i$ and $b_1, b_2, ..., b_i$ are the coefficients characterizing the recursive filter.

Adversarial Attacks. It has been shown that a small perturbation on machine learning model input can lead to model misclassification (Szegedy et al., 2014). Such adversarial attacks are also feasible for ASR systems.

Specifically, given an input audio waveform x, by adding an intentionally crafted small perturbation δ, the adversary can cause the ASR system to produce a wrong transcription $f(x + \delta) \neq f(x)$. In the audio domain, *untargeted* adversarial attacks are usually less interesting than *targeted* attacks as pointed out by researchers in (Carlini & Wagner, 2018; Qin et al., 2019). In a targeted attack, a target transcription y is pre-selected such that a crafted (small) perturbation δ can cause the ASR system to produce the target transcription.

Unlike attacks in the computer vision domain that are relatively well studied, attacks on ASRs are only explored very recently. There has not been a standard metric for measuring perturbation. In (Carlini & Wagner, 2018), researchers adapted the l_∞ metric from the image processing domain to bound the magnitude of audio perturbation. However, as reported in (Schönherr et al., 2019), audio perturbation with small l_∞ is perceptible to humans. In order to make adversarial perturbation stealthy, psychoacoustic models are leveraged (Schönherr et al., 2019; Qin et al., 2019). The idea of using psychoacoustic models is that a loud signal can mask other signals at nearby frequencies (Bosi & Goldberg, 2002). Other metrics are also explored by researchers (Liu et al., 2019). Our proposed approach can detect adversarial attacks regardless of these measurement metrics used in adversarial audio generation.

3 Method

Adversarial attacks modify natural audio signals by applying small perturbation. Existing attacks use either l_p norm or psychoacoustic model to construct adversarial audio. According to our study of adversarial audios in the frequency domain in Appendix B, the introduced perturbation by l_p norm based attacks almost evenly distributes over the entire frequency and time scope, which is similar to ambient noise. As such, recursive filters constitute effective counter-measure by their nature. The perturbation by psychoacoustic model has substantial presence in the high frequency. Recursive filters have the nice ability of “flattening” high frequency signals. Hence, the (malicious) features encoded by the perturbation can be largely corrupted.

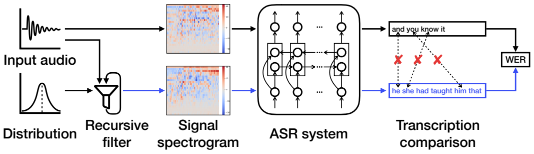


Figure 1: Architecture of FIRM.

We devise an adversarial audio detection technique FIRM based on recursive filters. Its workflow is illustrated in Figure 1. It consists of two simultaneous process paths. The path on top is the standard prediction procedure of an ASR system. Given an input audio waveform, it is first transformed to the frequency domain. The resulted signal spectrogram is subsequently fed to the ASR system. Finally the corresponding transcription is produced by the ASR. The path on the bottom differs from the one on top at signal pre-processing. The input audio waveform passes through a recursive filter before being transformed to signal spectrogram. FIRM compares the transcribed sentences from both paths using the *word error rate* (WER) metric. If an input audio exhibits a large WER between the two result transcriptions, it is considered adversarial.

Although existing audio attacks rely on high frequency signal, rendering recursive filters an effective defense method, future attacks may focus on low frequency signal. We will show in Appendix D that an adaptive attack that is aware of the presence of our defense, particularly its coefficients, can successfully evade the defense. Therefore, to provide protection for low frequency signal, we leverage the observation that *natural audio signal is not sensitive to various RF settings while adversarial signal is* (see Appendix C). That is, different RF settings (coefficients) yield similar transcription for natural audio signal but diverse transcription for adversarial signal as the latter tends to be much less robust due to its stealthiness requirement, namely, the perturbation has to be small.

The constitution of RF allows natural integration of randomization. Specifically, when computing each output signal value, a set of coefficients is sampled from some distribution as follows.

$$\begin{aligned} y_i = c_0 x_i + c_1 x_{i-1} + c_2 x_{i-2} + \ldots + c_{\lfloor k/2 \rfloor} x_{i-\lfloor k/2 \rfloor} \\ + c_{\lfloor k/2 \rfloor+1} y_{i-1} + c_{\lfloor k/2 \rfloor+2} y_{i-2} + \ldots + c_{k-1} y_{i-\lfloor k/2 \rfloor}, \end{aligned} \tag{2}$$

where $\sum_{i=0}^{k-1} c_i = 1, c_i \sim \mathcal{D}$. Coefficients $c_0, c_1, \ldots, c_{k-1}$ are sampled from a distribution $\mathcal{D}$ and normalized with the sum of 1. Theoretically, $\mathcal{D}$ can be any distribution. The results of employing different distributions are discussed in Appendix F. Unless otherwise stated, we use the uniform distribution $U(0, 1)$ and window size $k = 5$ as the default setting.

Note that the randomization in FIRM is by its nature different from the random local noise in (Carlini & Wagner, 2018), which has shown that such noise is not effective in improving the robustness of ASR systems. Specifically, our randomization has global effect through the recursive filter.

4 Evaluation

The evaluation is conducted on two end-to-end trained automatic speech recognition (ASR) systems and four state-of-the-art adversarial attack methods. In addition to exiting known attacks, strong

Table 1: Evaluation of detecting adversarial audios for different approaches. TD is the state-of-the-art detection approach proposed by Yang et al. (2019). RF denotes vanilla recursive filter. FIRM represents randomized RF. WS (Liu et al., 2019), CW (Carlini & Wagner, 2018) and PsyImp/PsyRob (Qin et al., 2019) are different adversarial attacks. TP and FP denote true positive rate and false positive rate respectively. TH represents the threshold of WER for determining TP and FP. AUC is the area under the ROC curve.

Detector	DeepSpeech						Lingvo								
	WS			CW			CW			PsyImp			PsyRob		
	TH	TP/FP (%)	AUC	TH	TP/FP (%)	AUC	TH	TP/FP (%)	AUC	TH	TP/FP (%)	AUC	TH	TP/FP (%)	AUC
TD	0.74	90.91/0.00	0.9959	0.80	60.90/20.00	0.6974	0.59	98.80/2.01	0.9881	0.59	98.73/1.80	0.9885	0.29	34.90/9.78	0.5778
RF	0.90	100.00/0.00	1.0000	0.79	98.10/ 5.00	0.9906	0.69	100.00/0.00	1.0000	0.69	100.00/0.00	1.0000	0.16	79.39/9.65	0.8785
FIRM	0.90	**100.00/0.00**	**1.0000**	0.93	**95.30/ 5.70**	**0.9724**	0.90	**99.90/0.00**	**1.0000**	0.90	**99.89/0.00**	**1.0000**	0.64	**87.10/5.69**	**0.9650**

adaptive attacks are performed on the detection mechanism to evaluate the robustness of defensive approaches. We also empirically compare FIRM with the state-of-the-art defense (Yang et al., 2019). Experimental results are discussed in the following. Please see Appendix E for detailed setup.

4.1 Detection of Adversarial Audios

Detection of Existing Attacks. The experimental results on existing attacks are shown in Table 1 (see ROC curves in Figure 5 in Appendix). We can observe that RF (using a vanilla recursive filter) and FIRM (using a randomized RF) consistently outperform the state-of-the-art defense technique TD (Yang et al., 2019). Particularly, TD performs the worst on the PsyRob attack, which is an advanced attack that leverages psychoacoustic model in the frequency domain. The detection AUC value of TD on DeepSpeech model reported in (Yang et al., 2019) is 0.936, which is much higher than that in our experiment. The reason is that Yang et al. (2019) only generated 50 adversarial samples with 3 target transcriptions, whereas we generate 1,000 adversarial audios with 10 target transcriptions. FIRM has similar performance with RF on all the attack scenarios except PsyRob. The PsyRob attack constructs more robust adversarial audios by utilizing room simulation. Vanilla RFs have pre-defined coefficients and mostly serve as a low-pass filter (with little protection on low frequency). FIRM, on the other hand, introduces randomness and hence provides protection on low frequency as well as high frequency. Also observe from the FP columns that FIRM has reasonable FPs, consistently better than TD. The TH column denotes the WER threshold used to determine adversarial audios. Similar to existing techniques (Xu et al., 2018), a threshold is selected for each attack scenario with the goal of minimizing false positives such that the normal functionalities of ASRs are not affected. WER is defined as $\frac{S+D+I}{N}$, where S, D, I are the numbers of substitutions, deletions and insertions between a subject sentence and a reference sentence, and N is the total number of words in the reference sentence. A large WER indicates substantial difference. Observe that FIRM uses the largest WER among all three methods, indicating adversarial audios become quite different after processed by randomized RF as shown in Table 3 in Appendix.

Detection of Adaptive Attacks. The results of adaptive attacks on vanilla RFs and FIRM are presented in Table 2. It can be observed that the AUC score of RF drops from 0.99-1.00 to 0.62-0.76. It demonstrates that in the adaptive settings, a vanilla RF can be evaded. FIRM, on the other hand, still has a high detection rate with an AUC score larger than 0.96.

Table 2: Evaluation of detecting adaptive adversarial attacks.

Detector	DeepSpeech			Lingvo					
	CW			CW			PsyImp		
	TH	TP/FP	AUC	TH	TP/FP	AUC	TH	TP/FP	AUC
RF	0.63	25/10	0.622	0.13	70/5	0.711	0.13	75/5	0.761
FIRM	0.88	**94/ 8**	**0.963**	0.88	**100/0**	**1.000**	0.88	**99/0**	**1.000**

5 Conclusion

We study the characteristics of audio signals in the frequency domain. We propose to use recursive filters and incorporate randomness to detect adversarial attacks. Our proposed approach FIRM can effectively detect attacks with AUC scores greater than 0.96 on five different attack scenarios and is resilient to adaptive attacks. FIRM outperforms the state-of-the-art defense technique.

ACKNOWLEDGEMENT

We thank Bai Li for the discussion of constructing adaptive attacks, and Zhisheng Hu for the assistance of conducting several experiments. This research was supported, in part by NSF 1901242 and 1910300, ONR N000141712045, N000141410468 and N000141712947, and IARPA TrojAI W911NF-19-S-0012. Any opinions, findings, and conclusions in this paper are those of the authors only and do not necessarily reflect the views of our sponsors.

REFERENCES

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pp. 284–293, 2018.

Marina Bosi and Richard E. Goldberg. *Introduction to Digital Audio Coding and Standards*. Kluwer Academic Publishers, Norwell, MA, USA, 2002. ISBN 1402073577.

Nicholas Carlini and David Wagner. Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pp. 1–7, 2018.

Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David Wagner, and Wenchao Zhou. Hidden voice commands. In *25th USENIX Security Symposium (USENIX Security 16)*, pp. 513–530, 2016.

Moustapha Cisse, Yossi Adi, Natalia Neverova, and Joseph Keshet. Houdini: Fooling deep structured prediction models. *arXiv preprint arXiv:1707.05373*, 2017.

Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Li Chen, Michael E Kounavis, and Duen Horng Chau. Adagio: Interactive experimentation with adversarial attack and defense for audio. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 677–681. Springer, 2018.

Yuan Gong and Christian Poellabauer. Crafting adversarial examples for speech paralinguistics applications. *arXiv preprint arXiv:1711.03280*, 2017.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep Speech: Scaling up End-to-End Speech Recognition. *arXiv preprint arXiv:1412.5567*, 2014.

Warren He, James Wei, Xinyun Chen, Nicholas Carlini, and Dawn Song. Adversarial example defense: Ensembles of weak defenses are not strong. In *11th USENIX Workshop on Offensive Technologies (WOOT 17)*, 2017.

Xiaolei Liu, Xiaosong Zhang, Kun Wan, Qingxin Zhu, and Yufei Ding. Towards Weighted-Sampling Audio Adversarial Example Attack. *arXiv preprint arXiv:1901.10300*, 2019.

Mozilla. Project DeepSpeech, 2017. URL `https://github.com/mozilla/DeepSpeech`.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR Corpus Based on Public Domain Audio Books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

Yao Qin, Nicholas Carlini, Ian Goodfellow, Garrison Cottrell, and Colin Raffel. Imperceptible, Robust, and Targeted Adversarial Examples for Automatic Speech Recognition. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 5231–5240, 2019.

Krishan Rajaratnam and Jugal Kalita. Noise flooding for detecting audio adversarial examples against automatic speech recognition. In *2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 197–201. IEEE, 2018.

Krishan Rajaratnam, Kunal Shah, and Jugal Kalita. Isolated and ensemble audio preprocessing methods for detecting adversarial examples against automatic speech recognition. *arXiv preprint arXiv:1809.04397*, 2018.

Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. Adversarial Attacks Against Automatic Speech Recognition Systems via Psychoacoustic Hiding. In *Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS)*, 2019.

Jonathan Shen, Patrick Nguyen, Yonghui Wu, Zhifeng Chen, Mia X Chen, Ye Jia, Anjuli Kannan, Tara Sainath, Yuan Cao, Chung-Cheng Chiu, et al. Lingvo: A Modular and Scalable Framework for Sequence-to-Sequence Modeling. *arXiv preprint arXiv:1902.08295*, 2019.

Steven W Smith et al. The Scientist and Engineer's Guide to Digital Signal Processing. 1997.

Liwei Song and Prateek Mittal. Inaudible voice commands. *arXiv preprint arXiv:1708.07238*, 2017.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing Properties of Neural Networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Weilin Xu, David Evans, and Yanjun Qi. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In *Proceedings of the 25nd Annual Network and Distributed System Security Symposium (NDSS)*, 2018.

Zhuolin Yang, Bo Li, Pin-Yu Chen, and Dawn Song. Characterizing Audio Adversarial Examples Using Temporal Dependency. In *International Conference on Learning Representations (ICLR)*, 2019.

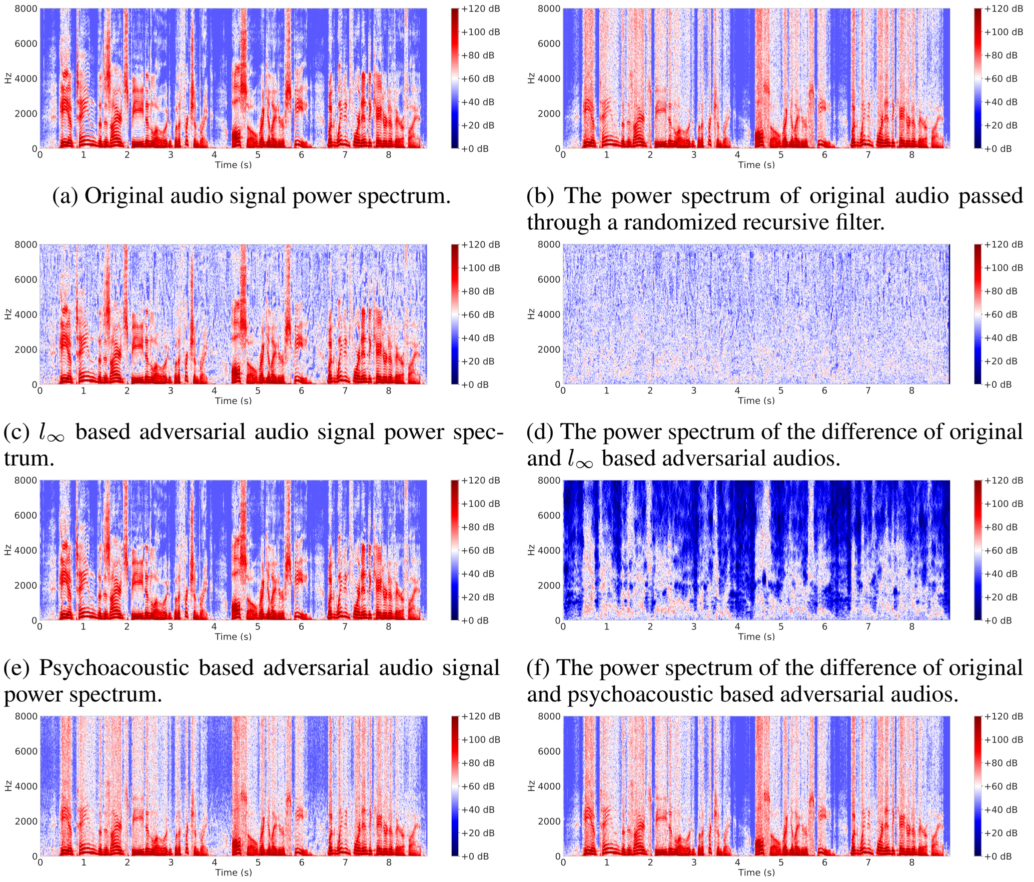
Qiang Zeng, Jianhai Su, Chenglong Fu, Golam Kayas, Lannan Luo, Xiaojiang Du, Chiu C Tan, and Jie Wu. A multiversion programming inspired approach to detecting audio adversarial examples. In *2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pp. 39–51. IEEE, 2019.

Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. Dolphinattack: Inaudible voice commands. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 103–117. ACM, 2017.

Appendix

A Related Work

In (Gong & Poellabauer, 2017; Cisse et al., 2017), researchers constructed *untargeted* adversarial audios that can cause incorrect outputs of ASRs. In (Carlini et al., 2016; Zhang et al., 2017; Song & Mittal, 2017), techniques were proposed to synthesize completely new audio to attack ASRs. In (Schönherr et al., 2019), researchers attacked a traditional ASR system called Kaldi, which is not trained end-to-end as DeepSpeech or Lingvo. Recent adversarial attacks proposed by Carlini & Wagner (2018); Liu et al. (2019); Qin et al. (2019) were conducted on state-of-the-art end-to-end trained systems. These attacks aim to generate adversarial audio that can make ASRs produce a target transcription by adding small perturbation to the original audio. To mitigate such attacks, Zeng et al. (2019) proposed to use output transcription from multiple ASRs; Rajaratnam & Kalita (2018) added simple noise with pre-defined frequency to audios for attack detection; Rajaratnam et al. (2018); Das et al. (2018) leveraged compression methods such as MP3 compression. While these techniques are effective for some attacks, they are less effective in other (more sophisticated) attacks. A state-of-the-art approach (Yang et al., 2019) utilized the observation that adversarial audio tends to have strong dependence across different parts such that removing a part causes substantial transcription difference. We use the approach as our baseline and show that it fails on a set of advanced attacks in Section 4.



(a) Original audio signal power spectrum.

(b) The power spectrum of original audio passed through a randomized recursive filter.

(c) l_∞ based adversarial audio signal power spectrum.

(d) The power spectrum of the difference of original and l_∞ based adversarial audios.

(e) Psychoacoustic based adversarial audio signal power spectrum.

(f) The power spectrum of the difference of original and psychoacoustic based adversarial audios.

(g) The power spectrum of l_∞ based adversarial audio passed through a randomized recursive filter.

(h) The power spectrum of psychoacoustic based adversarial audio passed through a randomized recursive filter.

Figure 2: Power spectra of input audio signals. Colors are used to denote the magnitude of signal at a specific frequency and a time point (as indicated by the color bar on the right-hand side).

B Frequency Analysis of Adversarial Audios

Adversarial attacks modify natural audio signals by applying small perturbation. Existing attacks use either l_p norm or psychoacoustic model to construct adversarial audio. The human perception of perturbation introduced by the l_p norm based adversarial attacks is similar to that of ambient noise. Figure 2 shows a set of *power spectra* of audio signals. The x axis of a power spectrum denotes the time; the y axis the frequency. The color of a point at (x, y) denotes the signal magnitude at the given timestamp and frequency. Note that the relation between color and magnitude is shown on the right as a color bar, with red denoting strong signal and blue weak signal. Figure 2d demonstrates the difference between an l_∞ based adversarial audio (Figure 2c) and the original audio (Figure 2a). Observe that the introduced perturbation almost evenly distributes over the entire frequency and time scope and hence resembles ambient noise. As such, recursive filters constitute effective countermeasure by their nature. Adversarial attacks that use psychoacoustic modeling aim to disguise perturbation in the original strong signal in the frequency domain, regardless of the frequency of the signal. For example, the injected signal using psychoacoustic modeling in Figure 2f has a shape similar to the original signal Figure 2a. Observe that the perturbation has substantial presence in the high frequency. Recursive filters have the nice ability of "flattening" high frequency signals. Hence, the (malicious) features encoded by the perturbation can be largely corrupted. Observe in Figure 2b, Figure 2g and Figure 2h that denote signals after filtering, high-frequency signals are flattened with indistinguishable differences, whereas low-frequency signals are mostly retained.

C Example to Demonstrate the Effect of Randomized RF

We feed a natural input audio that has gone through our randomized RF into the ASR to obtain the transcribed sentence. We then compare the transcription to that without the filter. We also perform the same experiment for signals generated by the attacks in (Carlini & Wagner, 2018) (CW) and (Qin et al., 2019) (PsyImp). The results are presented in Table 3. To reduce non-determinism caused by randomization, we ran the ASR with the filter 10 times for each signal and present the most dominating one. The others have negligible differences (e.g., one word difference).

Observe that for the normal input audio, the difference is negligible (word "less" becomes "left"), after processed by the filter. Adversarial audios are substantially affected by the filter. Also observe that the transcribed sentences of filter-processed adversarial audios highly resemble the normal transcription, suggesting FIRM may allow an ASR to recover from attacks (to some extent).

Table 3: Examples of normal audios (Normal), adversarial audios generated by Carlini & Wagner (2018) (CW) and by Qin et al. (2019) (PsyImp) transcribed by the ASR. The input audios are passed through the filter (✓) or not (✗) before feeding into the ASR.

Input	Filter	Transcription
Normal	✗	the more she is engaged in her proper duties the less leisure will she have for it even as an accomplishment and a recreation
	✓	the more she is engaged in her proper duties the left leisure will she have for it even as an accomplishment and a recreation
CW	✗	old will is a fine fellow but poor and helpless since missus rogers had her accident
	✓	more she is engaged in her proper duties the last measure will she have her it even as an accomplishment and a recreation
PsyImp	✗	old will is a fine fellow but poor and helpless since missus rogers had her accident
	✓	the more human gaits in her proper duties the left leisure will she have her it even as an accomplishment and the recreation

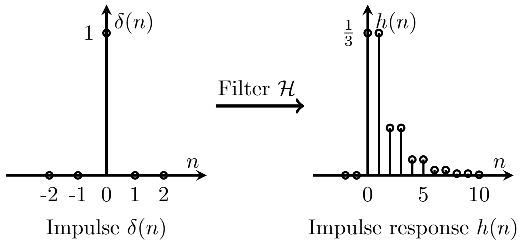


Figure 3: Example of impulse response. The left figure denotes an impulse signal $\delta(n)$. The right represents the corresponding impulse response $h(n)$ by a filter $\mathcal{H}$.

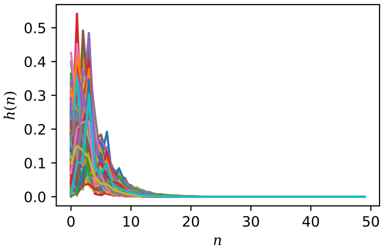


Figure 4: Impulse responses of randomized BRFs.

D Adaptive Attacks

Impulse Response. To understand the characteristics of a filter, the *impulse response analysis* is commonly leveraged. In this work, impulse response analysis is used in constructing adaptive adversarial attacks. Given a special input signal called *impulse*, the impulse response is the output produced by the filter. It characterizes the behavior of a filter. Formally, an input impulse $\delta(n)$, with $n \in \mathbb{Z}$, is defined as follows.

$$\delta(n) = \begin{cases} 1 & n = 0, \\ 0 & n \neq 0. \end{cases} \tag{3}$$

The impulse response $h(n)$ is the output of a filter $\mathcal{H}$ given the impulse $\delta(n)$, which is also denoted as $h(n) \triangleq \mathcal{H}\{\delta(n)\}$. To analyze the effect of a filter $\mathcal{H}$ on an input sequence $x(n)$, we can represent the input sequence as a weighted sum of the impulse, i.e.,

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k), k \in \mathbb{Z}. \tag{4}$$

Then the output sequence $y(n)$ can be denoted as

$$\begin{aligned} y(n) = \mathcal{H}\{x(n)\} &= \mathcal{H}\left\{\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right\} && (5) \\ &= \sum_{k=-\infty}^{\infty} x(k)\mathcal{H}\{\delta(n-k)\} && \text{(linearity of } \mathcal{H}\text{)} \\ &= \sum_{k=-\infty}^{\infty} x(k)h(n-k). && \text{(time-invariant of } \mathcal{H}\text{)} \end{aligned}$$

The key is that output $y(n)$ is the convolution of $x(n)$ and $h(n)$, i.e., $y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$. We use the following simple example to demonstrate the impulse response of a recursive filter.

Example. Suppose we have a recursive filter $\mathcal{H}$ with unknown behaviors and we want to identify its output function $y(n)$ given an input sequence $x(n)$. Through impulse response analysis, we obtain the response function $h(n) = (\frac{1}{3})^{\lfloor n/2 \rfloor + 1}$ for $n \geq 0$ and $h(n) = 0$ for $n < 0$. Applying the convolution sum in Equation 5 yields $y(n) = \sum_{k=-\infty}^{n} (\frac{1}{3})^{\lfloor k/2 \rfloor + 1} x(n-k)$ for $n \geq 0$, which is simplified to $y(n) = \frac{1}{3}x(n) + \frac{1}{3}x(n-1) + \frac{1}{3}y(n-2)$. Figure 3 shows the impulse response of the filter.

Adaptive Attack Construction. Existing adversarial attacks are conducted on the original ASR systems without any defense mechanism. It is more interesting to investigate how attackers adapt to bypass known defense/detection approaches. We analyze how a vanilla recursive filter and a randomized RF are constructed so as to design more powerful attack methods.

Recall that recursive filters leverage long-term history to compute the current signal value as discussed earlier. We use the same example to illustrate how to construct attack on recursive filters defended ASRs. Suppose we have a recursive filter as follows.

$$y(n) = \frac{1}{3}x(n) + \frac{1}{3}x(n-1) + \frac{1}{3}y(n-2), \tag{6}$$

where $x(n)$ is the input signal at time step n and $y(n)$ is the output signal. Due to the recursive nature of the filter, Equation 6 can be written in an explicit form by (recursively) expanding the output term $y(n-2)$ on the right-hand side.

$$y(n) = \frac{1}{3}x(n) + \sum_{k=0}^{n-1}(\frac{1}{3})^{n-k}x(k). \tag{7}$$

With the above explicit form, each output signal can be directly computed using all the previous input signal. The coefficient associated with each input signal is $(\frac{1}{3})^{n-k}$. It is straightforward to obtain all the coefficients beforehand, which constructs an input-output coefficient matrix with the size of $O(n^2)$. For a 10-second long audio waveform, it comprises 146800 input signals. That is, there are at least $\frac{1}{2} \cdot 146800^2 \approx 1.08 \times 10^{10}$ nonzero entries in the coefficient matrix. The computation for one pass (of filtering) is limited. When applying gradient-based attacks, Expectation over Transformation (Athalye et al., 2018) is commonly utilized for optimization. The computation, however, will be extremely expensive, as each forward (prediction) pass and each backward (gradient calculation) pass require the multiplication of the coefficient matrix and the input signal. To reduce the computation cost introduced by the large matrix multiplication, we resort to leveraging the exponentially decreasing property of recursive filters. As shown in Figure 3, the impulse response of the example recursive filter has large values around the origin and decreases exponentially with the increase of x-axis value. That is, the impact of previous long-term signals is limited/negligible. More specifically, input signal values are integers ranging from -2^{15} to $2^{15}-1$. Assume $\eta = \sum_t a_t$ (where $a_t > 0$) is the sum of all the remaining coefficients associated with the previous long-term signals, then $\eta \times (2^{15}-1) < 1$ can hence be disregarded. This characteristic of recursive filters offers the opportunity to utilize a more local filter for cost-efficient computation. We substitute the recursive filter in Equation 6 with a local filter with coefficients calculated using Equation 7 but not in η, i.e.,

$$y(n) = \sum_{k=0}^{20}(\frac{1}{3})^{\lfloor k/2 \rfloor + 1} \cdot x(n-k) \tag{8}$$

In Section 4, we demonstrate that with this equivalent substitution, regular recursive filters can be bypassed under adaptive settings.

Regarding our proposed randomized RF (defined in Equation 2), the coefficients associated with each input signal are sampled from some distribution. That is, there does not exist an explicit form as Equation 7 for these filters. All the calculation between input signals and output signals have to be conducted recursively, which introduces extremely high computation cost for gradient-based adaptive attacks. We hence leverage an analytic procedure. We sample 1,000 randomized RFs and conduct the impulse response analysis on these filters. The results are visualized in Figure 4. Observe that most impulse response values concentrate on the left part of the figure (x-axis less than 20). This showcases that there are only a limited number of input samples which have noticeable influence on the output values. Based on this observation, we approximate a randomized RF using a local filter with a finite number of coefficients as is done for vanilla filters. The results of adaptive attacks on these filters are represented in Section 4. The results show that randomized RFs cannot be evaded due to its random nature. Here, we use an approximate approach to performing the attack due to the extremely high computation cost. There is a possibility that our proposed approach can be evaded if the recursive filters and the introduced randomness can be better modeled. We plan to further explore this in future work.

E Experiment Setup

Two well-known ASR systems, two widely used speech-to-text datasets, and four state-of-the-art adversarial attacks are used for evaluating the effectiveness of different detection approaches.

Table 4: Statistics of attacks (WS (Liu et al., 2019), CW (Carlini & Wagner, 2018) and PsyImp/PsyRob (Qin et al., 2019)) on two ASR systems (DeepSpeech and Lingvo). Succ. denotes the attack success rate. WER is the word error rate of feeding adversarial audios to ASRs. Time shows the efficiency of different attacks (measured with GPU usage).

Metric	DeepSpeech		Lingvo		
	WS	CW	CW	PsyImp	PsyRob
Succ. (%)	100.00	100.00	99.60	94.70	33.49
WER (%)	0.00	0.00	0.30	2.49	65.52
Time[*](d)	-[†]	10×4	14×4	14×4	9×4

[*] The attack generation time was calculated based on #days × #GPUs.
[†] Adversarial audios generated by WS were downloaded from the original paper's website.

ASR Systems. We use two ASR systems, DeepSpeech (Hannun et al., 2014) and Lingvo (Shen et al., 2019). DeepSpeech is an open source speech-to-text engine implemented by Mozilla (Mozilla, 2017) based on the model proposed in (Hannun et al., 2014). Lingvo is a sequence-to-sequence model with attention based on the Listen, Attend and Spell model (Shen et al., 2019).

Datasets. Two datasets are employed: Mozilla Common Voice and LibriSpeech (Panayotov et al., 2015). The former is a large open source dataset containing thousands of hours of recorded audios. We use the same input set in (Carlini & Wagner, 2018) consisting of 100 audios. For each audio a, we randomly choose 10 transcriptions from the produced transcriptions of the other 99 audios as the target transcription. In other words, we use the attack methods to generate 10 adversarial audios from a, each causing the ASRs to produce the 10 respective target transcriptions. This is the same attack setting used in (Carlini & Wagner, 2018). LibriSpeech is a corpus with thousands of hours of English speech. We follow the convention and randomly select 1000 samples as the source audios, and then randomly select 1000 transcriptions different from the ones of the source audios as the target transcriptions. Each target transcription has similar length as the original transcription.

Adversarial Attacks. Four different adversarial attacks are considered in the evaluation. The first attack (CW) is the l_∞ attack in (Carlini & Wagner, 2018). It was originally applied to DeepSpeech using the CTC loss. We additionally perform the attack on Lingvo. We also use the *weighted-sampling* (WS) attack in (Liu et al., 2019) that enhances CW in efficiency and robustness by focusing on specific parts of distortion with adjusted weights. The third attack (Qin et al., 2019) leverages psychoacoustic modeling to generate human-imperceptible adversarial audios (PsyImp). The fourth is its extension that constructs robust over-the-air adversarial audios (PsyRob). Table 4 presents the statistics of the four attacks. For the WS attack, since there is no available implementation, we downloaded the generated adversarial audios published online[1] by the authors. For CW and PsyImp, we use the same setting as in the original papers (Carlini & Wagner, 2018; Qin et al., 2019), which consists of 1,000 adversarial audios for each attack. For the PsyRob attack, we randomly choose 200 audio samples from the LibriSpeech dataset for evaluation, which is twice the size in the original setting (Qin et al., 2019). The generated adversarial audios are tested under 100 random room configurations, which comprises 20,000 test combinations. As we use a larger set of audio samples to perform the attack, the attack success rate is slightly lower than that reported in the original paper (Qin et al., 2019) but still reasonable. Most these attacks take days or weeks to generate all the adversarial audios as shown in Table 4. Other than existing adversarial attacks, we further perform adaptive attacks. We make CW and PsyImp attacks adaptive as they have high attack success rates. We randomly select 100 audio samples to perform these attacks.

F Design Choices

Various distributions can be used to generate the randomized RFs in FIRM. We study three well-known distributions, namely, gamma distribution, normal distribution, and uniform distribution, as well as the mixture of these three. We use fixed parameters for these distribution: Gamma $\Gamma(1,1)$, Normal $\mathcal{N}(0,1)^2$, and Uniform $U(0,1)$. The mixture utilizes a discrete uniform distribution with

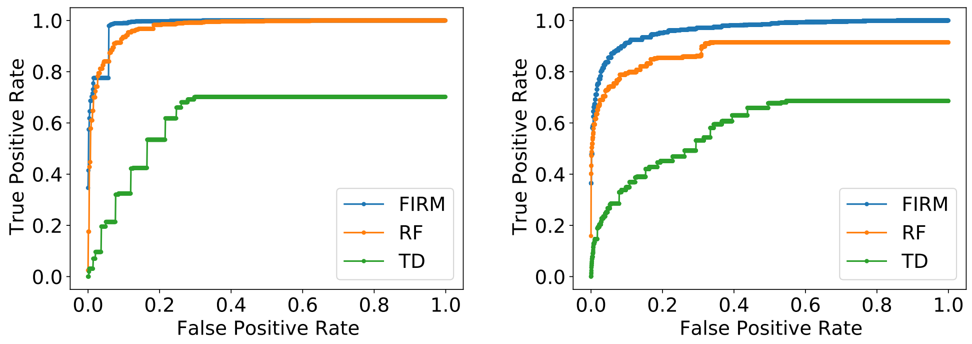
[1] `https://sites.google.com/view/audio-adversarial-examples/`

[2] The folded normal distribution is in fact used as coefficients are designed with positive values.

three values, where each value represents one of the above three distributions. For each output signal y_i, a distribution is selected to sample the coefficients. In the design of FIRM, a fixed window size is employed for the output signal computation. We also study how different window sizes affect the performance. Table 5 demonstrates the AUC scores of FIRM with various configurations. Interestingly, almost all the configurations have AUC scores more than 0.9 under various attack scenarios. FIRM with a larger window size tends to have lower scores. We believe that a larger window size creates dependence between the current signal and a distant past, which leads to unstable output. Different distributions have indistinguishable effect on the detection performance.

Table 5: Detection performance (AUC score) of FIRM with different configurations.

Dist.	Size	DeepSpeech		Lingvo		
		WS	CW	CW	PsyImp	PsyRob
Gamma	3	0.9917	0.9821	1.0000	1.0000	0.9574
	5	0.9917	0.9402	0.9999	0.9999	0.9767
	7	0.9917	0.9126	0.9996	0.9995	0.9656
	9	0.9917	0.8711	0.9986	0.9985	0.9391
	11	0.9917	0.8254	0.9970	0.9967	0.8913
Normal	3	1.0000	0.9874	1.0000	1.0000	0.9393
	5	0.9917	0.9607	1.0000	1.0000	0.9762
	7	0.9917	0.9234	0.9999	0.9998	0.9759
	9	0.9917	0.8942	0.9990	0.9990	0.9561
	11	0.9835	0.8674	0.9983	0.9981	0.9252
Uniform	3	1.0000	0.9914	1.0000	1.0000	0.9215
	5	1.0000	0.9724	1.0000	1.0000	0.9650
	7	0.9917	0.9463	1.0000	1.0000	0.9795
	9	0.9917	0.9151	0.9997	0.9997	0.9656
	11	0.9917	0.8931	0.9989	0.9988	0.9466
Mixture	3	1.0000	0.9870	1.0000	1.0000	0.9450
	5	0.9917	0.9587	0.9999	0.9999	0.9748
	7	0.9917	0.9300	0.9999	0.9998	0.9730
	9	0.9917	0.8951	0.9987	0.9987	0.9541
	11	0.9917	0.8603	0.9976	0.9974	0.9228



(a) Detecting CW attack on DeepSpeech

(b) Detecting PsyRob attack on Lingvo

Figure 5: The ROC curves for different detection approaches