Defending Against Imperceptible Audio Adversarial Examples Using Proportional Additive Gaussian Noise

Ethan Mendes
Westford Academy

Kyle Hogan
MIT

July 2020

Abstract

Neural networks are susceptible to adversarial examples, which are specific inputs to a network that result in a misclassification or an incorrect output. While most past work has focused on methods to generate adversarial examples to fool image classification networks, recently, similar attacks on automatic speech recognition systems have been explored. Due to the relative novelty of these audio adversarial examples, there exist few robust defenses for these attacks. We present a robust defense for inaudible or imperceptible audio adversarial examples. This approach mimics the adversarial strategy to add targeted proportional additive Gaussian noise in order to revert an adversarial example back to its original transcription. Our defense performs similarly to other defenses yet is the first randomized or probabilistic strategy. Additionally, we demonstrate the challenges that arise when applying defenses against adversarial examples for images to audio adversarial examples.

1 Introduction

Automatic speech recognition (ASR) technology has been incorporated into the lives of people around the globe. Providing advantages in both convenience and accessibility, so-called *digital assistants* are present in the majority of personal devices. Users can issue verbal commands to their devices via the digital assistant which will parse their speech using ASR, allowing the intended function to be executed by the device. This has facilitated hands-free device interaction and is used by Amazon Alexa [5], Apple Siri [6], Google Home [1], and Microsoft Cortana [2] to do everything from answering calls and playing music to shopping online and managing a smart home system.

These digital assistants, as well as all other ASR systems, employ machine learning to process and transcribe the user's speech. Common ASR implementations such as the Lingvo ASR system [47] and DeepSpeech [23] use a deep-learning based speech classification approach for Speech-to-Text transcription. While this approach provides high accuracy for speech recognition, it introduces vulnerabilities in the form of adversarial examples.

Adversarial examples are maliciously formed inputs to a machine learning (ML) algorithm that are *misclassified*, i.e. they are recognized differently by the algorithm than by a human user, despite differing only slightly from a correctly classified input [9, 50]. Widely studied in the space of image recognition [7, 11, 12, 14, 18, 19, 22, 29, 34–37, 39–41, 46], adversarial examples are considered to be inherent to ML with all classification algorithms being susceptible to some degree [45].

Recent attacks have shown successful generation of adversarial examples targeting ASR systems [4,13,15,21,25, 27,42,48,53,54,56,57]. These attacks seek to be imperceptible to human listeners while simultaneously causing the execution of malicious commands by their devices. Critically, as many people listen to audio content from music, podcasts, and videos at home, an attack such as that of Carlini and Wagner that embeds adversarial commands into music or speech files has easy access to a user's devices [13]. An overt version of this was demonstrated in a 2017 episode of the television show *South Park* in which viewers' Amazon Alexa devices were triggered with commands to add items to their shopping lists [26]. These concerns are exacerbated with the rise of untrustworthy and unverified media sharing platforms such as YouTube or Podcasts.

While some current attacks perform poorly when played over the air to be picked up by a device's microphone [13,44], this is not an inherent limitation of the technique and should not be relied upon for security [42, 54]. Similarly, the basic access control provided by speaker recognition is an insufficient defense. Speaker recognition is itself susceptible to adversarial examples and favors usability over robustness, recognizing a broad range of speakers beyond just an individual user [28].

Defenses have been proposed specifically against AAEs that seek to identify or eliminate adversarial perturbations to an input [17, 43, 55]. These are typically easily surmountable by slight modifications to an attacker's strategy or severely limit the accuracy of the ASR system they seek to protect. For example, mitigation methods have used input transformations such as downsampling (reducing the audio sampling rate). However, these defenses were shown to be ineffective against adaptive attacks which are specifically designed to target a particular defense [55]. Other defenses that have been tried in practice have attempted to detect AAEs. One such method attempted to distort a potential adversarial per-

turbation using preprocessing techniques and then calculate the error between the transcription of the distorted sample and the original. The sample could be deemed adversarial based on where the error fell in regards to a precomputed threshold [52]. However, these detection methods are heuristic in that the aforementioned value of threshold is based on the training dataset and the adversarial generation method making them hard to extend.

Recently, in the space of image recognition, a group of defenses for adversarial examples for images coined *certified defenses* have been shown to be able to provide rigorous guarantees against these attacks [16, 30, 31].

We show it is difficult to modify, generalize or extend these certified defenses for adversarial examples on images to defenses for AAEs. This difficulty stems from the fact that all the certified defenses that currently exist take advantage of the adversarial strategy utilized by many vision attacks of adding uniform low magnitude adversarial perturbations. However, this adversarial strategy is trivially different because an adversary can take advantage of principles of psychoacoustics, the scientific study of human sound perception [32], while generating AAEs. Psychoacoustics can help the adversary inject adversarial noise into the specific regions of the audio that are inaudible to humans creating **imperceptible audio adversarial examples** [42, 44, 51]. Thus, unlike the adversarial strategy countered by certified defenses, imperceptible AAEs are generated by adding localized high magnitude adversarial perturbations.

We craft an effective defense specifically for imperceptible AAEs by accounting for this different adversarial strategy and considering the specific regions of audio targeted by these attacks.

We utilize the psychoacoustic property of auditory masking to inject Gaussian noise in selected regions of the perturbed input audio having the greatest masking thresholds and thus the highest probability of containing high magnitude adversarial perturbation. This defense method proves to be effective because perturbations injected during the defense are added in the same manner and in the same relative locations with respect to frequency as adversarial perturbations.

Based on the nature of our defense, it is possible for an adversary to get around it by adding adversarial perturbation to audible regions of the audio. However, in doing this, the adversary is sacrificing imperceptibility for adversarial potency. Therefore, with the implementation of our defense, any adversarial example will either have limited effectiveness or be able to be detected by a human.

Finally, we have found that this defense prevents adversarial misclassifications, while maintaining correctness on non-adversarial data and allowing the ASR system to recover from such an attack. While our defense performs similarly to other defense strategies, it is the first probabilistic defense for imperceptible audio adversarial examples.

2 Background and Definition

2.1 Speech Recognition

In this paper we focus on speech-to-text ASR systems which take speech as input and output a transcription of its contents. We let $f(\cdot)$ denote the ASR system. Thus, in the case of $f(x) = y$, the speech input to the ASR system, x, provides a transcription y.

2.2 Adversarial Examples

An adversary can create an adversarial example, x' by adding an adversarial perturbation, δ_A, to the input, x: $x' = x + \delta_A$. An adversarial example is created such that $f(x') = y'$, where $y' \neq y$. An adversary may try to optimize for a specific preset value of y'. Such adversarial examples are referred to as **targeted**. To create targeted adversarial examples, an adversary would solve the following optimization problem:

$$\begin{aligned} &\underset{\delta_A}{\text{argmin}}\ l(f(x+\delta_A), y') \\ &\text{for } ||\delta_A|| < \epsilon \end{aligned}$$

where ϵ is the maximum allowed magnitude of the adversarial perturbation. Variations of this loss function have been used in many attacks such as those by Szegedy et al. and Carlini and Wagner [13, 50].

2.3 l_p norms

Many attacks and defenses have used the l_p norm to constrain the magnitude of the adversarial perturbation [30, 44]. These norms are denoted as $||\cdot||_p$ and can be formally defined as:

$$||x||_p = (\sum_{i=1}^{n} |x_i|^p)^{1/p}$$

where x is a vector. For example, l_2 norm, or the Euclidean distance norm, is commonly used in vision attacks to spread the adversarial perturbation throughout the image, rendering it imperceptible to humans.

2.3.1 Frequency Masking Threshold

The field of psychoacoustics attempts to understand and model the human perceptibility of audio. The natural audibility of humans can be approximated by a global frequency masking threshold. All signals that fall below the threshold are imperceptible to humans. This threshold is calculated by approximating the effect of maskers, or relative high magnitude signals. These maskers can "mask out" or render imperceptible adjacent signals if they fall under the calculated local masking threshold attributed to the masker. By compiling the effects of all the local masking thresholds in the frequency domain, a global masking threshold can be calculated for each frame of the audio.

We denote this global threshold as $\theta_x(s, \nu)$, where s denotes the spectrum of a particular frame and ν denotes the frequency.

These frequency masking thresholds have practical applications such as MP3 compression, in which every imperceptible signal that falls below the masking threshold is removed [24]. However, just because humans cannot hear the these masked out signals, does not mean that an ASR system does not use them for classification. In fact, studies have found that classification accuracy on MP3 compressed samples, especially for low bit-rates, was significantly lower than on uncompressed audio [8].

2.3.2 Calculating the Masking Threshold

The frequency masking threshold $\theta_x(s, \nu)$ can be calculated through the following equation:

$$\theta_x(s, \nu) = 10^{\text{Quiet}(\nu)} + \sum_{i=1}^{N_m} 10^{T[b(\nu), b(i)]}$$

Here, Quiet(ν) refers to the universal absolute threshold in quiet, i.e. the threshold that models the least intensity sounds that humans can sense for every frequency ν. Additionally, N_m refers to the number of maskers, while $T[b(\nu), b(i)]$ refers to the precomputed masking effect of the masker located at $b(i)$ on the maskee located at $b(\nu)$. $b(\cdot)$ refers to the Bark index that corresponds to a certain frequency. Note that the Bark scale is a method to measure frequency that is motivated by the field of psychoacoustics. Please refer to the work by Lin and Abdulla or the paper by Qin et al. for a more detailed explanation of the masking threshold calculation [33, 42].

2.4 Imperceptible Audio Adversarial Examples

An adversary can use the frequency masking threshold during the optimization process to hide the adversarial perturbation below the masking threshold so that it is inaudible to humans. Such adversarial examples are deemed **imperceptible**. These attacks can be generated by adding another term to the minimized loss function that accounts for the inaudibility of the adversarial perturbation. For example, Qin et al. use the following loss function:

$$\underset{\delta_A}{\text{argmin}}\ l_{net}(f(x + \delta_A, y') + \alpha \times l_\theta(x, \delta_A)$$

Here, l_{net} is a cross entropy loss function used to create an adversarial example x' that fools the ASR system into making the targeted prediction y'. Additionally, l_θ is a loss function computed with hinge loss that constrains the adversarial perturbation, δ_A, below the calculated masking threshold of x, θ_x. Note that α is an adaptive factor that is controls the imperceptibility of the adversarial example, i.e. for larger values of α a larger proportion of the total loss can be attributed to l_θ.

The attack is split into two iterative stages. During the first stage, the adversarial perturbation is optimized to hit the targeted transcription ($\alpha = 0$). In the second stage, the adversary attempts to maintain high adversarial accuracy while trying to constrain the adversarial perturbation below the masking threshold (α is tweaked until convergence).

See Section 3 and the work by Qin et al. for more details about their attack [42].

3 Related Works

Prior work has provided methods to generate audio adversarial examples, strategies for detection and weak defense for AAEs, and robust defense methods for adversarial examples on images. However, to the our knowledge, we present the first specific robust defense for imperceptible AAEs.

3.1 Attacks

3.1.1 Primitive and Norm Bounded Attacks

Zhang et al. presented one of the first untargeted AAE generation algorithms, DolphinAttack. This method added ultrsonic inaudible adversarial perturbations, which had a frequency under 20 kHz. Additionally, these attacks can be played over the air [57]. However, they could be trivially defended against using a high-pass filter. Carlini and Wagner were able to generate targeted adversarial examples from any input audio (including music) using Connectionist Temporal Classication (CTC) loss and the l_∞ norm to quantify the amount of adversarial perturbation added. They were also able to generate imperceptible, albeit impractical, AAEs by targeting silence. However, their AAEs do not remain potent when played over the air [13].Yakura et al. generate AAEs that are targeted and robust when played over the air, but cannot be trivially defended against like the DolphinAttack. However, these AAE introduce high magnitude adversarial perturbations and are only effective for short phrases consisting of two or three words. They use the l_2 norm to quantify adversarial perturbation [54].

3.1.2 Imperceptible Audio Adversarial Examples

Schönherr et al., Qin et al. and Szurley and Kolter diverged from the work of Carlini and Wagner and Yakura et al., which had used the l_p metric to measure and restrict the magnitude of adversarial perturbation added.

Instead, in these approaches, the principle of psychoacoustic hiding was used to deliberately inject adversarial perturbation under the masking threshold of human perceptibility. AAEs generated with the approach of Schönherr et al. attacked hybrid Deep Neural Network-Hidden Markov Model (DNN-HMM) based ASR systems such as Kaldi [44], while Qin et al. and Szurley and Kolter attacked Recurrent Neural Network (RNN) based ASR

systems such as the Lingvo ASR system and DeepSpeech. Unlike, Schönherr et al., the AAEs generated by Qin et al. and Szurley and Kolter can also be played over the air [42, 51].

We consider the attacks by Qin et al. and Szurley and Kolter to be state of the art, and choose to use the former for evaluation.

3.2 Detection and Defense Methods

3.2.1 Mitigation Methods for Audio Adversarial Examples

Additionally, rudimentary prepossessing defense methods have been proposed to mitigate the effect of AAEs. For example, applying MP3 compression to AAEs to remove all signals below the human perceptibility threshold has been proposed as a defense method for imperceptible AAEs [17]. However, this method results in decreased classification accuracy on benign samples. Additionally, Subramanian et al. showed that MP3 compression is less effective than additive white noise in defending against audio adversarial examples [49]. Furthermore, some attacks have been able to generate adversarial examples that are robust to MP3 compression by accounting for the compression during the optimization process [13]. Additionally, Yang et al. also explored the effect of simple input transformations such as quantization (rounding the amplitude of sampled audio to the nearest integer multiple of a predefined constant) and downsampling (reducing the audio sampling rate) They found that these methods were successful in preventing adversarial misclassifications. However, these defenses were only able to recover the benign transcription for 63.8% of AAEs, and they moderately impacted the classification accuracy of benign samples. Also, these these transformations were not able to defend against adaptive attacks and were not tested on imperceptible AAEs [55].

3.2.2 Detection Methods for Audio Adversarial Examples

Methods to detect adversarial examples have also been extensively studied. Yang et al. took advantage of the property of temporal dependence (correlations between successive waveform segments) of audio to detect AAEs by only transcribing a portion of the audio and comparing the transcription to a transcription of the whole audio using word error rate (WER) [55]. Rajaratnam et al. detect AAE by calculating a flooding score defined as the amount of random noise that needs to be added to change the classification of particular audio input. If a flooding score falls below a predetermined threshold, the input is determined to be an AAE [43]. Tamura et al. use various processing techniques in an attempt to remove the adversarial perturbation. They then compare the transcription of the original input and the scrubbed input and deem the input adversarial if the character error rate (CER) between the two above a predetermined threshold. Although these methods are fairly successful in detecting adversarial examples, they do not allow an ASR system to recover from an AAE. Additionally, a study by Carlini et al. on the strength of defense and detection methods against adaptive attacks showed that the method proposed by Yang et al. [55] required only a slightly higher magnitude adversarial perturbation to be overcome than the preprocessing defenses mentioned above [12]. Unfortunately, Carlini et al. did not evaluate the detection methods devised by Rajaratnam et al. or Tamura et al. but similar results can be expected because they all use a threshold optimized through experimental data. Also, the empirical basis of these thresholds means that it cannot be broadly applied to all ASR systems and AAE generation algorithms.

3.2.3 Defense Methods for Vision Attacks

However, defenses for image adversarial examples have been demonstrated to be effective. Certified defenses, which are grounded in information theory, ensure the robustness of the defense for up to a certain threshold of adversarial perturbation added to the original input image. For example, PixelDP inserts a noise layer, which contains neurons whose values are randomly sampled from Gaussian or Laplacian distributions, into the classification network during both training and testing. This work quantifies adversarial perturbation through the l_1 and l_2 norms [30]. Cohen et al. add onto the work of Lecuyer et al. by developing tighter robustness certificates for the l_2 norm case [16]. While these defenses are promising methods to mitigate adversarial examples generated on images, their use of l_p norms cannot adapt well to imperceptible AAE, which are based in psychoacoustics [42].

4 Outline of Defense

In this section we present our method to defend against imperceptible AAEs. The basic premise of this defense is to add a defensive perturbation (δ_D) to raw inputs before they are fed into the ASR system. However, unlike the certified defenses for images [16, 30, 31], the defensive perturbation added at each frequency band is proportional to the value of the masking threshold for each frequency. This defensive strategy is effective because it allocates defensive perturbation dynamically based on relative magnitudes of adversarial perturbation at each frequency. The full defense generation process is outlined in Algorithm 1 and depicted in Figure 1.

4.1 Prepossessing the Raw Input

In order to calculate the masking threshold, we must transform the data to the frequency domain. To do this, we use the same approach as Qin et al. by applying a short time Fourier transform (STFT) to the input audio originally in the time domain (Figure 1.a) with a window

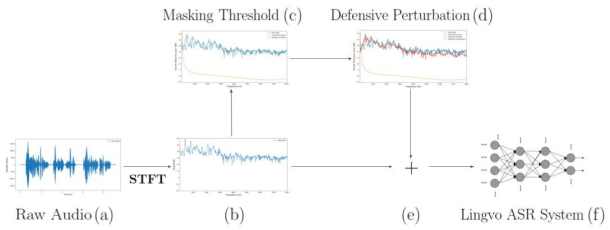

Figure 1: **Overview of the defense process:** Stage (a) consists of the raw speech in the time-domain. In Stage (b), this raw audio is transformed into the frequency domain using a Short-time Fourier transform (STFT). During Stage (c), this transformed audio is used to calculate the frequency masking threshold (green curve). In Stage (d), the defensive perturbation (red curve) is calculated from the frequency masking threshold. The outputs of Stage (b) and (d) are summed and serve as the input to the ASR system in Stage (e). Finally, this input is fed through the trained ASR system (neural network) to render a final transcription.

size of 2048 samples and a hop size (the number of samples by which the Hann window is shifted) of 512 samples [42]. This produces a spectra of s frames, where s depends on the duration of the input audio, each in the frequency domain (Figure 1.b). Because we are using a window size of 2048 for the STFT, the frame size is 1025.

4.2 Calculating the Masking Threshold

We calculate the global masking threshold of each audio waveform using the same method as Qin et al. This method is presented in the background section. The result is the array $\theta_x(s, \nu)$ which contains the global masking thresholds at each frequency ν of the spectrum of each frame s (Figure 1.c). For specifics on how to calculate the masking threshold please refer to the attack paper by Qin et al. [42].

4.3 Calculating the Defensive Perturbation

As previously mentioned, the defensive perturbation, $\delta_D(s, \nu)$, for a particular audio sample is based on the audio masking threshold of that sample (Figure 1.d). To do this, for each frequency value of each frame of the original audio after a STFT, we sample from a Gaussian distribution with the standard deviation being a multiple of the threshold amplitude value for that bin and frequency value, $\theta_x(s, \nu)$:

$$\delta_D(s, \nu) = max(0, N(\mu, \sigma))$$

We use the max function here to ensure that the amplitude is positive. The values of μ and σ control the magnitude of the defensive perturbation. Because the amount of noise added at a specific frequency is dependent on the masking threshold value at that frequency, μ and σ are functions of the masking threshold $\theta_x(\nu)$.

After trying various distributions, we found that the best results are achieved by setting $\mu = 3 \times \sigma$ as:

$$\mu := 3k \times \theta_x(s, f)$$
$$\sigma := k \times \theta_x(s, f)$$

where k denotes the proportionality. We choose this relationship between μ and σ as an example because, due to the empirical rule of statistics, there is a less than 0.5% probability that the sample from the Gaussian distribution is less than 0. Thus, this relationship allows us to avoid consistently adding a 0 defensive perturbation due to the use of the max function. This allows us to establish a baseline as we can gauge the full effect of accuracy due to Gaussian noise. For example, for $k = \frac{1}{6}$ the region under the masking threshold is flooded with our defensive perturbation, which is the same region where the adversarial perturbation is most likely to be added.

4.4 Apply Defensive Perturbation

We can apply the defensive perturbation $\delta_D(s, \nu)$ to the original audio after STFT to form the defensively perturbed audio (Figure 1.e): $x_D(s, f)$:

$$x_D(s, f) = x(s, f) + \delta_D(s, f)$$

Finally note that we only modify the amplitudes of the input, thus, the phase of x_D is the same as that of x.

Algorithm 1 Application of Defense

Input: audio waveform in the frequency domain: $x(s, \nu)$, calculated masking threshold of the audio waveform: $\theta_x(s, \nu)$, set of bins: S_b, set of frequencies in each bin: S_ν, proportionality factor: k
Notation: bin: b, frequency: ν, defensive perturbation: $\delta_D(s, \nu)$, mean: $\mu(s, \nu)$, standard deviation: $\sigma(s, \nu)$, defensively perturbed audio: $x_D(s, \nu)$
for b in S_b **do**
  for ν in S_ν **do**
    $\sigma(s, \nu) \leftarrow k \times \theta_x(s, \nu)$
    $\mu(s, \nu) \leftarrow 3k \times \theta_x(s, \nu)$
    $\delta_D(s, \nu) \leftarrow N(\mu(s, \nu), \sigma(s, \nu))$
    $x_D(s, \nu) = x(s, \nu) + \delta_D(\nu)$
  end for
end for

5 Experimental Evaluation

Here we will present the results of experiments that measure the effectiveness of our defense in terms on adversarial effectiveness, benign accuracy, and recovery against imperceptible AAEs.

5.1 Evaluation Dataset and Attack

5.1.1 LibriSpeech Dataset

For experimentation, we use the LibriSpeech dataset, which is a freely available and downloadable corpus of 1000 hours of human speech sampled at a rate of 16 kHz [38].

5.1.2 Iterative Imperceptible Adversarial Example Generation Algorithm

We evaluate our defense on the state-of-the-art adversarial examples generated by Qin et al. that are built to be imperceptible [42]. This is a two stage iterative attack. The first stage focuses on fooling the ASR system into making the targeted prediction, while the second stage focuses on decreasing the perceptibly of the adversarial examples. This attack is build using Lingvo, which is a framework used to build sequenced-based Tensorflow models [47]. Please refer to Section 2 and 3 for the specific of the attack.

5.2 Evaluation Metrics

5.2.1 Word Error Rate

We employ the word error rate metric (WER) to evaluate our defense and compare its efficacy to other defense strategies. This metric calculates a percentage error between the ground truth, intended transcription, and the hypothesis, actual transcription returned by the ASR system. The word error rate is calculated as: $WER = \frac{S+D+I}{N}$, where S, D, and I are the number of substitutions, deletions, and insertions between the intended and actual transcription, and N is the number of words in the intended transcription. Generally, the higher the WER the greater the difference between the intended and actual audio transcription. However, for our results, three specific cases are important:

- $WER = 0\%$ In this case the number of substitutions, insertions and deletions is 0, indicating that the intended transcription perfectly matches the actual transcription.
- $WER = 100\%$ This case occurs when the ASR system does not return a transcription, and thus, $S + D + I = N$
- $WER > 100\%$ Here, the WER exceeds 100% when the intended transcription is vastly different, and usually shorter, than the actual transcription.

5.2.2 Measurements of Interest

When evaluating our defense, we measure the following:

- Adversarial-Adversarial WER: This is the WER between the adversarially targeted transcription and the actual transcription of the adversarial examples returned by the ASR system. We desire high values for this measurement as this would indicate that the adversary is unsuccessful as they are not able to hit their targeted transcription.
- Benign-Benign WER: Because we cannot discriminate between adversarial and benign samples as inputs, we must apply our defense to all inputs. Thus, we need to preserve high accuracy on benign samples even after our defense is applied. This metric measures the WER between the intended benign transcription and the actual transcription of a benign sample returned by the ASR system. The lower the WER, the higher the benign accuracy.
- Adversarial-Benign WER: Finally, we wish to understand how well our defense reverts the actual transcriptions of adversarial examples back to the original transcription before the adversarial perturbation was added. To do this, we calculate the WER between the actual transcription of adversarial examples returned by the ASR system to their corresponding benign transcriptions. Lower values of this WER indicate a greater reversion to the original transcription.

5.3 Finding an Optimal k

In this section, we use the values of the mean and standard deviation of the Gaussian distribution from which the defensive perturbation is sampled of $3k \times T(b, \nu)$ and $k \times T(b, \nu)$ respectively. Because larger values of k indicate that our defense adds more random noise added to the input and vice versa, we can experiment with different values of k to find the optimal magnitude of the defensive perturbation. The results are shown in Figure 2.

For very low values of k ($\log(k) \approx 9$), note that the adversary is highly successful, as the Adversarial-Adversarial WER is close to zero, indicating that the adversarially targeted transcription meets the actual transcription of the adversarial example. From the figure, it is also clear that there are three distinct regions. The first is in the range $-9 < \log(k) < -6$, where the Adversarial-Adversarial WER begins to increase. In the second region ($-6 < \log(k) < -3$), the Adversarial-Adversarial WER and the Adversarial-Benign WER plateau at 100%. Recall that when $WER = 100\%$ the ASR system renders no transcription. Thus, in this region, the adversary's perturbation is interacting with our defensive perturbation to fool the ASR system into not detecting any intelligible speech. Finally, the third region ($-2 < \log(k) < 0$) is optimal region because in this region we achieve a high Adversarial-Adversarial WER, and low Benign-Benign and Adversarial-Benign WER. Note that when $\log(k) > 0$ the Benign-Benign WER and Adversarial-Adversarial WER begin to increase. We have found that this increase

Defense	Adversarial-Adversarial (WER)	Benign-Benign (WER)	Adversarial-Benign (WER)
MP3	132.44%	6.41%	14.58%
Quan-256	134.59%	4.86%	9.96%
Ours ($k = -1.50$)	135.30%	3.09%	19.42%

Table 1: WER comparisons between defense strategies

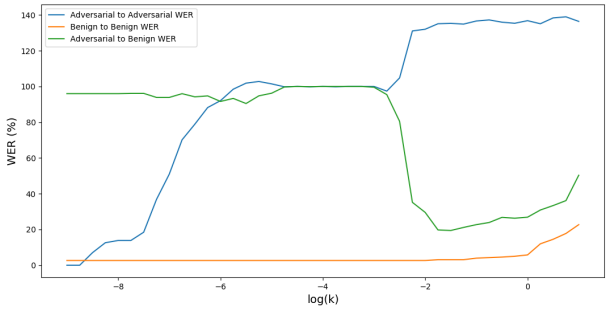

Figure 2: **Optimizing k:** The effect of manipulating k is clear from this graph. The optimal region for k, which increases the Adversarial-Adversarial WER and decreases both the Benign-Benign WER and the Adversarial-Benign WER, is $-2 < \log(k) < 0$

is due to the presence of too much noise from the defensive perturbation in the system such that the accuracy of the ASR system steadily decreases.

5.4 Comparison to Existing Defenses

We compare the results of our defense to those of MP3 compression and quantization, the most successful defensive method used by Yang et al. Through the process of quantization, all values of the sampled audio are rounded to the nearest multiple of some constant q. To be consistent with Yang et al. we set $q = 256$, and denote this defense as Quan-256 [55]. The results are shown in Table 1.

By these results, it is clear that our defense is comparable to those of Quan-256 and MP3 compression. For example, our defense maintains benign accuracy (low Benign-Benign WER) and disrupts the transcription of the adversarial example (high Adversarial-Adversarial WER) more so than the Quan-256 and the MP3 compression defense. However, it performs worse than the other defenses in the reversion of transcriptions of adversarial examples back to the original transcriptions as shown by the higher Adversarial-Benign WER.

However, it is important to note that the quantization and MP3 compression defenses are deterministic processes and are thus easily able to be thwarted if the adversary has knowledge of the defense. An attack in such a case is called an **adaptive** adversarial example. For example, Yang et al. state that "all the adversarial audios can be resistant against quantization transformations and it only increased a small magnitude of adversarial perturbation, which can be ignored by human ear" [55]. Similarly, as mentioned, Carlini and Wagner, on which the Qin et al. attack used in this paper is based, developed an attack which was robust against MP3 compression by optimizing a modified loss function that accounts for the compression [13]. On the other hand, our defense is probabilistic, as we randomly sample from Gaussian distributions at each frequency value. Although this has not yet been tested experimentally, such a method is theorized to be more robust against adaptive attacks due to the results of the similar randomized certified defenses in the image domain [16,30,44] . See Section 7 for more information about how such attacks could be generated.

6 Discussion

Most previous defenses for adversarial examples, especially certified defenses, fail to represent the adversarial strategy in its full complexity. These defenses, use l_p norms to quantify the adversarial perturbation added, which heavily and unrealistically restricts the adversarial strategy. Thus, while the security guarantees presented by certified defenses that use l_p norms seem promising, the associated adversarial restrictions render such strategies ineffective and impractical.

The imperceptible audio adversarial examples that we study in this paper further prove that l_p will become obsolete as adversarial examples become more robust and imperceptible. However, using qualities of human perception during adversarial example generation is not limited to the audio domain. For example, in the future, an adversary might use the findings of the field of visual perception by using psychovisual factors such as luminance, lightness, and contrast to choose their adversarial perturbation [10]. In fact, work has been done to identify human perceptibility thresholds of images [3,20]. These adversarial examples would easily fool current defenses that use the l_p norm. Although such attacks do not yet exist for images, based on the rapid growth of the body of work under the umbrella of adversarial machine learning, these adversarial examples are inevitable.

Therefore, when creating a defense for these more robust adversarial examples, we need to be cognizant of the corresponding fields of psychological perception and the associated complex adversarial strategies. It is imperative that, defense strategies use latent qualities of the input data e.g. masking thresholds, as these are employed by smarter adversaries during the adversarial optimization process. In this work we make the first strides by demon-

strating how such a defense can be created by using the knowledge of the location of adversarial perturbations to directly target and mitigate its effect. In the future, we hope that similar defense strategies will follow in the fields of vision, natural language, etc.

7 Future Work

There are a few future directions for this project. First, in this current work, we choose to set the mean of the distribution from which we sample our defensive as three time the standard deviation. Parameter tuning, in which we optimize the values of this mean and standard deviation, could be used to achieve better results.

Additionally, we can generate adaptive adversarial examples in the future. Such adversarial examples will be created with knowledge of the k of our defense during generation process. To do this, the adversary uses Expectation over Transformation introduced by Athalye et al., where the expectation of the loss function is optimized over the set of potential transformations ($t \sim T$) [7]. In our case, T is equivalent to the Gaussian distribution from which we sample our defensive perturbation. Our defense is successful against these adaptive attacks attacks if they fail when our defense is applied or require a more audible perturbation to force the adversarially targeted prediction.

8 Conclusion

In this paper, we present a defense specifically designed to counter imperceptible adversarial examples. This defense strategy applies additive noise sampled from Gaussian distributions proportional to the masking threshold at each frequency value to the input to an automatic speech recognition system. This defense achieves similar accuracy to existing defenses, while also providing a probabilistic strategy that is difficult to counter even if the adversary has knowledge of the defense. Also, unlike previous defenses, this work chooses a realistic and robust adversarial strategy. Finally, this paper ushers in a new defensive strategy of mimicking the adversarial approach in a defense.

9 Acknowledgements

This collaboration was made possible through the MIT PRIMES program. I would specifically like to thank my mentor Kyle Hogan for her guidance.

References

[1] Google assistant. `https://developers.google.com/assistant`.

[2] Microsoft cortana dev center. `https://developer.microsoft.com/en-us/cortana/`.

[3] Allen, E., Triantaphillidou, S., and Jacobson, R. Perceptibility and acceptability of jpeg 2000 compressed images of various scene types. In *Image Quality and System Performance XI* (2014), vol. 9016, International Society for Optics and Photonics, p. 90160W.

[4] Alzantot, M., Balaji, B., and Srivastava, M. Did you hear that? adversarial examples against automatic speech recognition. *arXiv preprint arXiv:1801.00554* (2018).

[5] Amazon. Amazon alexa official site: What is alexa? `https://developer.amazon.com/en-US/alexa`.

[6] Apple. Siri - apple developer. `https://developer.apple.com/siri/`.

[7] Athalye, A., Engstrom, L., Ilyas, A., and Kwok, K. Synthesizing robust adversarial examples. *arXiv preprint arXiv:1707.07397* (2017).

[8] Barras, C., Lamel, L., and Gauvain, J. . Automatic transcription of compressed broadcast audio. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)* (2001), vol. 1, pp. 265–268 vol.1.

[9] Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. *Lecture Notes in Computer Science* (2013), 387–402.

[10] Bouman, C. A. The visual perception of images. `https://engineering.purdue.edu/~bouman/ece637/notes/pdf/Vision.pdf`, January 2020.

[11] Brown, T. B., Mané, D., Roy, A., Abadi, M., and Gilmer, J. Adversarial patch, 2017.

[12] Carlini, N., and Wagner, D. Towards evaluating the robustness of neural networks, 2016.

[13] Carlini, N., and Wagner, D. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)* (2018), IEEE, pp. 1–7.

[14] Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security* (New York, NY, USA, 2017), AISec '17, Association for Computing Machinery, p. 15–26.

[15] Cisse, M. M., Adi, Y., Neverova, N., and Keshet, J. Houdini: Fooling deep structured visual and speech recognition models with adversarial examples. In *Advances in neural information processing systems* (2017), pp. 6977–6987.

[16] Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing, 2019.

[17] Das, N., Shanbhogue, M., and Chen, S.-T. Compression to the rescue : Defending from adversarial attacks across modalities extended abstract.

[18] Elsayed, G., Shankar, S., Cheung, B., Papernot, N., Kurakin, A., Goodfellow, I., and Sohl-Dickstein, J. Adversarial examples that fool both computer vision and time-limited humans. In *Advances in Neural Information Processing Systems* (2018), pp. 3910–3920.

[19] Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D. Robust physical-world attacks on deep learning models, 2017.

[20] Girod, B. What's wrong with mean-squared error? In *Digital images and human vision*, A. Watson, Ed. MIT Press, Cambridge, Mass, 1993, ch. 15, pp. 207–219.

[21] Gong, Y., and Poellabauer, C. Crafting adversarial examples for speech paralinguistics applications. *CoRR abs/1711.03280* (2017).

[22] Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples, 2014.

[23] Hannun, A. Y., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., and Ng, A. Y. Deep speech: Scaling up end-to-end speech recognition. *CoRR abs/1412.5567* (2014).

[24] Information Technology – Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbits/s – Part3: Audio. Standard, International Organization for Standardization, Geneva, CH, 1993.

[25] Iter, D., Huang, J., and Jermann, M. Generating adversarial examples for speech recognition. *Stanford Technical Report* (2017).

[26] Jackson, C., and Orebaugh, A. A study of security and privacy issues associated with the amazon echo. *International Journal of Internet of Things and Cyber-Assurance 1*, 1 (2018), 91–100.

[27] Khare, S., Aralikatte, R., and Mani, S. Adversarial black-box attacks on automatic speech recognition systems using multi-objective evolutionary optimization, 2018.

[28] Kreuk, F., Adi, Y., Cisse, M., and Keshet, J. Fooling end-to-end speaker verification with adversarial examples. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2018), IEEE, pp. 1962–1966.

[29] Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533* (2016).

[30] Lecuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., and Jana, S. Certified robustness to adversarial examples with differential privacy, 2018.

[31] Li, B., Chen, C., Wang, W., and Carin, L. Certified adversarial robustness with additive noise, 2018.

[32] Lin, Y., and Abdulla, W. H. Principles of psychoacoustics. In *Audio Watermark*. Springer, 2015, pp. 15–49.

[33] Lin, Y., and Abdulla, W. H. Principles of psychoacoustics. In *Audio Watermark*. Springer, 2015, pp. 15–49.

[34] Liu, Y., Chen, X., Liu, C., and Song, D. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770* (2016).

[35] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks, 2017.

[36] Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., and Frossard, P. Universal adversarial perturbations, 2016.

[37] Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. Deepfool: a simple and accurate method to fool deep neural networks, 2015.

[38] Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2015), IEEE, pp. 5206–5210.

[39] Papernot, N., McDaniel, P., and Goodfellow, I. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016.

[40] Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security* (New York, NY, USA, 2017), ASIA CCS '17, Association for Computing Machinery, p. 506–519.

[41] Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. The limitations of deep learning in adversarial settings, 2015.

[42] Qin, Y., Carlini, N., Cottrell, G., Goodfellow, I., and Raffel, C. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *Proceedings of the 36th International Conference on Machine Learning* (Long Beach, California, USA, 09–15 Jun 2019), K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97 of *Proceedings of Machine Learning Research*, PMLR, pp. 5231–5240.

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

[44] Schönherr, L., Kohls, K., Zeiler, S., Holz, T., and Kolossa, D. Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. *CoRR abs/1808.05665* (2018).

[45] Shafahi, A., Huang, W. R., Studer, C., Feizi, S., and Goldstein, T. Are adversarial examples inevitable? *CoRR abs/1809.02104* (2018).

[46] Sharif, M., Bhagavatula, S., Bauer, L., and Reiter, M. K. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (New York, NY, USA, 2016), CCS '16, Association for Computing Machinery, p. 1528–1540.

[47] Shen, J., Nguyen, P., Wu, Y., Chen, Z., Chen, M. X., Jia, Y., Kannan, A., Sainath, T. N., Cao, Y., Chiu, C., He, Y., Chorowski, J., Hinsu, S., Laurenzo, S., Qin, J., Firat, O., Macherey, W., Gupta, S., Bapna, A., Zhang, S., Pang, R., Weiss, R. J., Prabhavalkar, R., Liang, Q., Jacob, B., Liang, B., Lee, H., Chelba, C., Jean, S., Li, B., Johnson, M., Anil, R., Tibrewal, R., Liu, X., Eriguchi, A., Jaitly, N., Ari, N., Cherry, C., Haghani, P., Good, O., Cheng, Y., Alvarez, R., Caswell, I., Hsu, W., Yang, Z., Wang, K., Gonina, E., Tomanek, K., Vanik, B., Wu, Z., Jones, L., Schuster, M., Huang, Y., Chen, D., Irie, K., Foster, G., Richardson, J., and et al. Lingvo: a modular and scalable framework for sequence-to-sequence modeling. *CoRR abs/1902.08295* (2019).

[48] Song, L., and Mittal, P. Inaudible voice commands, 2017.

[49] Subramanian, V., Benetos, E., and Sandler, M. B. Robustness of adversarial attacks in sound event classification.

[50] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).

[51] Szurley, J., and Kolter, J. Z. Perceptual based adversarial audio attacks, 2019.

[52] Tamura, K., Omagari, A., and Hashida, S. Novel defense method against audio adversarial example for speech-to-text transcription neural networks. In *2019 IEEE 11th International Workshop on Computational Intelligence and Applications (IWCIA)* (2019), pp. 115–120.

[53] Taori, R., Kamsetty, A., Chu, B., and Vemuri, N. Targeted adversarial examples for black box audio systems, 2018.

[54] Yakura, H., and Sakuma, J. Robust audio adversarial example for a physical attack. *arXiv preprint arXiv:1810.11793* (2018).

[55] Yang, Z., Li, B., Chen, P., and Song, D. Characterizing audio adversarial examples using temporal dependency. *CoRR abs/1809.10875* (2018).

[56] Yuan, X., Chen, Y., Zhao, Y., Long, Y., Liu, X., Chen, K., Zhang, S., Huang, H., Wang, X., and Gunter, C. A. Commandersong: A systematic approach for practical adversarial voice recognition, 2018.

[57] Zhang, G., Yan, C., Ji, X., Zhang, T., Zhang, T., and Xu, W. Dolphinatack: Inaudible voice commands. *CoRR abs/1708.09537* (2017).