Dompteur: Taming Audio Adversarial Examples with Psychoacoustic Compression

Anonymous Submission

Abstract

In light of current knowledge, adversarial examples for deep neural networks seem to be inevitable. These hand-crafted inputs, drastically change the output of learning systems and are critical in our digital society. For example, Automatic Speech Recognition (ASR) systems influence our daily lives, by serving as hands-free interfaces to our homes and cars. When applied against ASR systems, adversarial example are typically unintelligible and human listeners only comprehend the original input while being oblivious to the perturbation introduced by an adversary. Accepting that our systems are vulnerable to adversarial examples, we propose the following perspective: When human listeners are presented with a perturbed input, they should be able to perceive it as such, i. e., the input should be audibly changed.

By applying the principles underlying MP3 compression, we can remove semantically irrelevant information from the input and train a model that resembles human perception more closely. We implement our idea in a tool named Dompteur[1] and demonstrate that our augmented system, in contrast to an unmodified baseline, utilizes the perceptible ranges of humans. This successfully forces adversarial examples into the audible range, while using minimal computational overhead and preserving benign performance. Further, we investigate an *adaptive attacker* which actively tries to avoid our augmentations, again, demonstrating that adversarial examples are clearly perceivable. Finally, we substantiate our claims by performing a hearing test with humans.

1 Introduction

The advent of deep learning has changed our digital society. Starting from simple recommendation [1] or image recognition applications [2], machine-learning systems have evolved to solve and play games on par with humans [3–6], to predict protein structures [7], to identify faces [8], and to recognize speech at the level of human listeners [9]. These systems are now virtually ubiquitous and are being granted access to critical and sensitive parts of our daily lives. They serve as our personal assistants [10], unlock the doors of our smart homes [11], or drive our autonomous cars [12].

Given these circumstances, the discovery of *adversarial examples* [13] has had a shattering impact. These specifically crafted inputs, can completely mislead machine learning-based systems. First used for evasion attacks [14], they have later been extended to any model [15] and to targeted attacks [13] (i.e. forcing the output to a specific prediction).

Primarily studied for image recognition [13], adversarial attacks have been also transferred to the audio domain [16–20]. The most advanced attacks start from a harmless input signal and change the prediction of the model towards a target transcription, while simultaneously *hiding* their malicious intent in the inaudible audio spectrum.

To address such attacks, the research community has developed various defense mechanisms [21–26]. All of the proposed defenses—in the ever-lasting cat and mouse game between attackers and defenders—have subsequently been broken [27]. Recently, Shamir et al. [28] even demonstrated that, given certain constraints, we can expect to always find adversarial examples for our models.

Given these circumstances, we ask the following question: *When we accept that adversarial examples exists, what else can we actually do?* We propose a paradigm shift: Instead of preventing *all* adversarial examples, we accept the presence of *some*, but we want them to be audibly changed.

To achieve this shift, we take inspiration from the machine learning community which shed a different light on adversarial examples: Illyas et al. [29] interpret the presence of adversarial examples as a disconnection between human expectations and the reality of mathematical function trained to minimize an objective. We tend to think that machine learning models must learn meaningful features, e. g., a cat has paws. This is a human's perspective on what makes a cat a cat. In contrast, Illyas et al. demonstrate that image classifiers utilize so-called brittle features, which are highly predictive yet not recognizable by humans.

[1] The french word for tamer.

Recognizing this mismatch between human expectations and the reality of machine learning systems, we propose DOMPTEUR, a novel *Automatic Speech Recognition* (ASR) system which more closely resembles the human auditory system. We use psychoacoustic compression [30, 31] to simulating the human auditory system. To this end, we include an additional pre-processing step in the ASR pipeline that removes *inaudible* ranges from the input. The effects are twofold: (i) The ASR system learns a better approximation of the human perception during training, and (ii) an adversary is forced to place any adversarial perturbation into audible ranges.

In a series of experiments we prove that our model more closely models the human auditory system: We demonstrate that our ASR, in contrast to an unmodified baseline, focus on perceptible ranges of the audio signal. Following Carlini et al. [32], we depart from the lab settings predominantly studied in prior work. We assume a white-box attacker with real-world capabilities, i.e., we grant them full knowledge of the system, as well as, not restricting the amount of perturbations they are allowed to introduce. Further, we investigate an *adaptive* attacker, i.e., attacker who actively adapts their strategies based on the deployed defenses. We demonstrate that for our augmented ASR, the added perturbations are significantly higher than on the baseline and can be clearly perceived, while requiring practically no computational overhead and remaining accurate for benign inputs. Additionally, by conducting a user study with human listeners, confirming that these adversarial examples are easily distinguishable from benign audio samples.

In summary, we make the following key contributions:

- **Psychoacoustic Augmented ASR.** We utilize psychoacoustic compression to bring ASR systems more in line with human expectations. We demonstrate that a fully trained ASR indeed utilizes non-audible signals that are not recognizable by humans. Furthermore, by pre-processing the input, we combat existing adversarial attacks and destroy most of the introduced perturbations.

- **Evaluation against Adaptive Attacker.** In a realistic scenario, an attacker will adapt and actively factor in any employed defense mechanism. We show that in case of an adaptive attack scenario, where we grant the attacker complete control over the input, we can successfully force them into the audible range.

- **Listening test.** To study the real quality of adversarial examples, we perform a study with an extensive listening test. We demonstrate that the adversarial examples against our system are much more perceptible by humans.

To support further research in this area, we open source both our prototype implementation and our pre-trained models online at https://github.com/dompteur/dompteur.

2 Technical Background

In the following, we discuss the background necessary to understand our augmentation of the ASR system. For this purpose, we briefly introduce the key concepts of ASR and give an overview of adversarial examples. Since our approach fundamentally relies on psychoacoustic modeling, we also explain masking effects in human perception and, in particular, psychoacoustic compression.

2.1 Speech Recognition

ASR constitutes the computational core of today's voice interfaces. Given an audio signal, the task of an ASR system is to automatically transcribe any spoken content. For this purpose, traditionally, purely statistical models were used. They now have been replaced by modern systems based on deep learning methods [33–35], often in the form of hybrid neural/statistical models [36].

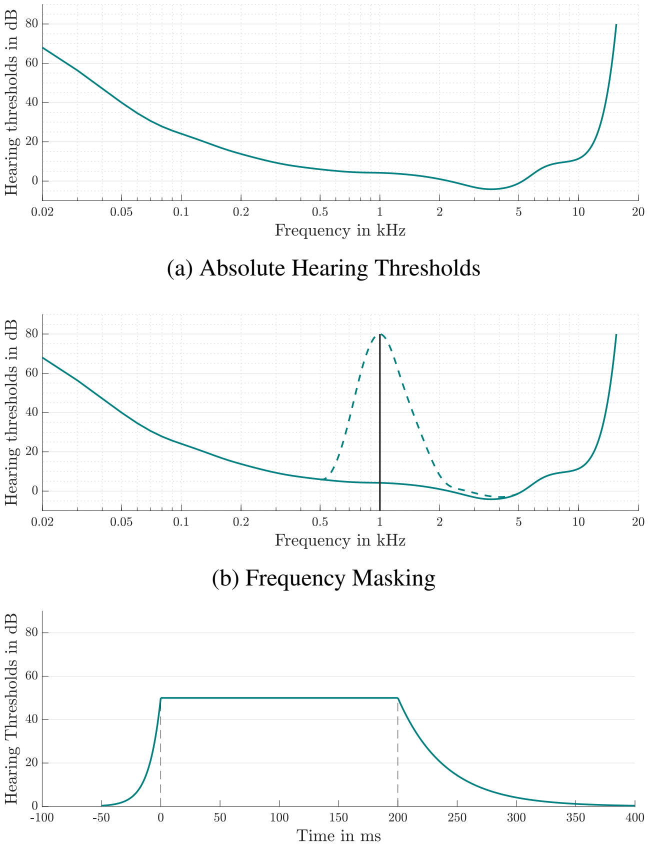
In this paper, we consider the open-source toolkit KALDI [37] as an example of such a modern hybrid system. It allows for substituting the previously statistical acoustic model by *Deep Neural Networks* (DNN), which are more flexible and less plagued by numerical issues for high-dimensional or correlated input data. This lets KALDI leverage recent advances in this field, and its high performance on many benchmark tasks has led to its broad use throughout the research community as well as in commercial products like e. g., Amazon's Alexa.

KALDI, and similar DNN/HMM hybrid systems, can generally be described as three-stage systems:

- *Feature Extraction.* For the feature extraction, a framewise discrete *discrete Fourier transform* (DFT) is performed on the raw audio data to retrieve a frequency representation of the input signal. The input features of the DNN are often given by the log-scaled magnitudes of the DFT-transformed signal.

- *Acoustic Model DNN.* The DNN acts as the *acoustic model* of the ASR system. From its DFT input features, it calculates the probabilities for each of the distinct speech sounds called *phones* of its trained language being present in each time frame. Alternatively, it may compute probabilities not of phones, but of so-called *clustered tri-phones* or of more general, data-driven units termed *senones*.

- *Decoding.* The output matrix of the DNN is used together with a *hidden Markov model* (HMM)-based language model to find the most likely sequence of words, i. e., the most probable transcription of the audio data. For this purpose, a dynamic programming algorithm, e.g., Viterbi decoding, is utilized to conduct a search for the



(a) Absolute Hearing Thresholds

(b) Frequency Masking

(c) Temporal Masking

Figure 1: **Psychoacoustic compression utilizes limitations of the human auditory system to compress audio files.** Figure 1a shows the average human hearing threshold in quiet. Figure 1b shows an example of masking, illustrating how a loud impulse at 1kHz shifts the hearing thresholds of nearby frequencies and Figure 1c shows how the recovery time of the auditory system after processing a loud signal leads to temporal masking.

best path through the underlying HMM, where the language model describes probabilities of word sequences and the acoustic model output gives the probability of being in each HMM state at each time.

2.2 Psychoacoustic Modeling

Recent attacks against ASR systems exploit intrinsics of the human auditory system to make adversarial examples less conspicuous [19, 38–40]. Specifically, these attacks utilize limitations of human perception to hide modifications of the input audio signal within inaudible ranges. For our approach, we use the same effects to *remove* all inaudible components from the input before processing:

- *Absolute Hearing Threshold.* Human listeners can only perceive sounds in a limited frequency range, in the best of cases between approximately 20 and 20,000 Hz, which diminishes with age. Moreover, for each frequency, the sound pressure is important to determine whether the signal component is in the audible range for humans. Measuring the *hearing thresholds*, i. e., the necessary sound pressures for each frequency to be audible in otherwise quiet environments, one can determine the so-called *absolute hearing threshold* as depicted in Figure 1a. Generally speaking, everything above the *absolute hearing thresholds* is perceptible in principle by humans, which is not the case for the area under curve. As can be seen, at the lower and higher frequencies, much more energy is required for a signal to be perceived. Note that the described thresholds only hold for cases where no other sound is present.

- *Frequency Masking.* The presence of another sound—a *masking tone*—can change the described *hearing thresholds* to cover a larger area. This *masking effect* of the masking tone depends on its sound pressure and frequency. Figure 1b shows an example of a 1 kHz masking tone, with its induced changes of the *hearing thresholds* indicated by the dashed line.

- *Temporal Masking.* Similarly to frequency masking, temporal masking is also caused by other sounds, but these sounds have the same frequency as the masked tone and are close to it in the time domain, as shown in Figure 1c. Its root cause lies in the fact that the auditory system needs a certain amount of time, in the range of a few hundreds of milliseconds, to recover after processing a higher-energy sound event to be able to perceive a new, less energetic sound. Interestingly, this effect does not only occur at the end of a sound but also, although much less distinct, at the beginning of a sound. This seeming causal contradiction can be explained with the processing of the sound in the human auditory system.

2.3 Adversarial Examples

Since the seminal papers by Szegedy et al. [13] and Biggio et al. [14], a field of research has formed around adversarial examples. The basic idea is simple: An attacker starts with a valid input to a machine learning system. Then, they add small perturbations to that input with the ultimate goal of changing the resulting prediction (or in our case, the transcription of the ASR).

More formally, given a machine learning model f and an input-prediction pair $\langle x, y \rangle$, where $f(x) = y$, we want to find a small perturbation δ s.t.:

$$x' = x + \delta \quad \wedge \quad f(x') \neq f(x).$$

In this paper, we consider a stronger type of attack, a targeted one. This has two reasons, the first being that an untargeted attack in the audio domain is fairly easy to achieve, and the second being that a targeted attack provides an actual real-life use case for adversarial examples. More formally, the attacker wants to perturb an input phrase x (i.e., an audio signal) with a transcription y (e.g., "Play the Beatles") in such a way that the ASR transcribes an attacker-chosen transcription y' (e.g., "Unlock the front door"). This can be achieved by computing an adversarial example x' based on a small adversarial perturbation δ s.t.:

$$x' = x + \delta \quad \wedge \quad ASR(x') = y' \quad \wedge \quad y \neq y'. \tag{1}$$

It exists a multitude of techniques for creating such adversarial examples. We introduce the method used by Schönherr et al. [19] since these form the basis for our evaluation in Section 4. The method can be divided into three parts:

In a first step, attackers choose a fixed output matrix of the DNN to maximize the probability of obtaining their desired transcription y'. As introduced in Section 2.1, this matrix is used in the decoding step of the ASR system to obtain the final transcription. They then, utilizes gradient decent to perturb a starting input x (i. e., an audio signal feed into the DNN), to obtain a new input x', which produces the desired matrix. This approach is generally chosen in a white-box attacks [18, 20, 41]. Note that we omit the feature extraction part of the ASR, however, Schönherr et al. have shown that this part can be integrated into the DNN itself [19]. A third (optional) step, is to utilize psychoacoustic hearing thresholds to restrict the added perturbations to the inaudible ranges. More technical details can be found in the original publication [19].

3 Psychoacoustics Augmentations

In this section we motivate and explain our design decision behind our proposed defense. From a high-level perspective, our goal is to modify the ASR system to model the human auditory system more closely. Precisely, we leverage psychoacoustic compression to limit the attacker's capabilities and force adversarial examples into human audible ranges.

3.1 Attacker Model

For a defense to achieve meaningful guarantees, we believe the attacker needs to be assumed to have *complete* control over the input. Following guidelines recently established by Carlini et al. [32] we embark from theoretical attack vectors and want to define a realistic threat model, capturing real-world capabilities of attackers. The key underlying insight is that the amount of perturbations caused by a real-world attack cannot be limited. This is easy to see; in the worst case, the attacker can always force the target output by replacing the input with the corresponding audio command.

In addition, previous works have successfully shown so-called parameter-stealing attacks, which build an approximation of a black-box system [15, 42–45]. Since an attacker has full control over this approximated model, they can utilize powerful white-box attacks against it, which transfer to the black-box model.

To summarize we use the following attacker model:

- *Attacker Knowledge:* Following Kerckhoffs's principle [46], we consider a *white-box* scenario where the attacker has complete knowledge of the system, including all model parameters, training data, etc.
- *Attacker Goals:* To maximize practical impact, we assume a targeted attack, i. e., the attacker attempts to perturb a given input x to fool a speech recognition system into outputting a false, *attacker controlled* target transcription y' based on Equation (1).
- *Attacker Capabilities:* The attacker is granted complete control over the input and we explicitly do not restrict them in any way on how δ should be crafted. Note, however, that δ is commonly minimized during computation according to some distance metric. For example, by measuring the *perceived* noise, an attacker might try to minimize the conspicuousness of their attack [19].
- *Adaptive Attacker:* We evaluate two different kind of attackers in this work. The first one (which we dub *static*), has no knowledge about our deployed defense The second one (which we dub *adaptive*), has complete knowledge about our deployed defense and actively tries to avoid it.

We choose this attacker model with the following in mind: We aim to limit the attacker not in the amount of applied perturbations, but rather confine the nature of perturbations itself. In particular, we want adversarial perturbations to be clearly perceptible by humans and, thus, strongly perturb the initial input such that the perturbation becomes audible for a human listener. In this case, an attack—although still viable—significantly loses its malicious impact in practice.

3.2 Approach

Our approach is based on the fact that the human auditory system uses only a subset of information of the raw audio signal to form an understanding of its content. In contrast, ASR systems are not limited to ranges of the given input. They can utilize even those not available in the human auditory perception. Consequently, for an attacker it is easy to hide changes within exactly those ranges. Therefore, the smaller the overlap between these two worlds, the harder it becomes for an attacker to maliciously add perturbations, while preserving the human understanding and thus evading detection.

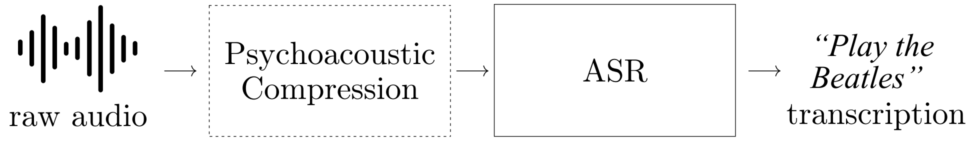


Figure 2: **High-level overview over our proposed augmentation of the ASR system.** Intuitively, we extend the ASR pipeline by an additional pre-processing step. During this pre-processing, the input is encoded using psychoacoustic compression, and subsequently decoded, to obtain an audio signal without the inaudible components.

To preclude that attack vector, we work towards bridging the gap between human and machine perception. Specifically, we leverage psychoacoustic phenomena, which are normally used for compressing audio files (cf. Section 2.2). In our case, we utilize the same techniques to process the input and remove all parts inaudible to humans. Intuitively, the compressed input still contains all relevant information both for machine transcription and for human listeners.

Thus, to augment a given ASR system with this approach, we add an additional *pre-processing* step as indicated in Figure 2. This pre-processing step is responsible for removing all inaudible parts from the input. Consequently, it affects the system both during training as well as run-time:

(i) *Training*. During training, the pre-processing creates a new training set which restricts the information set available to the ASR system to the subset also used by human listeners. Features derived from this information set are more aligned with human perception as we avoid a mismatch between information used for human understanding and speech recognition.

We demonstrate in Section 4.3 that the transcription quality is not degraded by this restriction.

(ii) *Run-time*. During run-time, the pre-processing step, again, removes all inaudible parts from the input files, forcing the attacker to place *any* modifications within audible ranges – perceptible to the human auditory system. In other words, an system augmented with this approach removes any hideout for the attacker and, consequently, forces the complete attack into the audible realm.

The basic principle has been discussed as a defense mechanism by Carlini and Wagner [41] and Rajaratnam et al. [47]. However, in both cases, it was solely used as a preprocessing step in order to destroy adversarial perturbations. Thus, it was neither evaluated in detail against an adaptive attack, nor if any adversarial examples were perceptible by humans.

3.3 Implementation

We built a prototype implementation of the proposed augmentation in a tool called Dompteur. Specifically, we apply our construction on the state-of-the-art ASR toolkit Kaldi and extend it with an additional pre-processing step that removes inaudible parts of the input (see Figure 2).

For the psychoacoustic compression, we choose the MPEG-2 layer III (MP3) codec and, in particular, the *LAME* encoder in version v3.100 [48]. The MP3 codec combines lossy compression by exploiting the perceptual limitations of the human auditory system to remove perceptually irrelevant parts of the audio signal together with lossless compression via a Huffman encoding scheme.

4 Evaluation

With the help of the following experiments, we empirically verify and assess our proposed approach according to the following four main aspects:

(i) *Benign Performance.* The augmentation of the system should retain performance on benign input. To verify this, we compare the accuracy for benign samples decoded with the original and augmented system.

(ii) *Static Attacker.* In a first step, we analyze the robustness of the augmented system against adversarial examples generated by a *static* attacker. This experiment assesses the efficiency of current, state-of-the-art white-box attacks against our defense.

(iii) *Adaptive Attacker.* In a realistic scenario, an attacker typically tries to *actively* factor in the defense. To evaluate against this kind of adversary, we analyze the robustness of our system against attacks that are aware of the employed defense mechanism.

(iv) *Listening Test.* Finally, we conclude with a listening test investigating the quality (i.e., the inconspicuousness) of the adversarial examples computed from the adaptive attacker against the augmented ASR system.

All experiments in this section were performed on a server running Ubuntu 18.04, with 128 GB RAM, an Intel Xeon Gold 6130 CPU, and four Nvidia GeForce RTX 2080 Ti. For our experiments, we use Kaldi in version 5.3 and train the system with the default settings from the *Wall Street Journal* (WSJ) training recipe. The corresponding WSJ-based speech corpus [49] contains approximately 80 hours of training data and consists of read sentences from the Wall Street Journal.

4.1 MP3 Compression

For the MP3 compression, we evaluate different levels of compression. These levels are specified in terms of a target bitrate, that (broadly speaking) describes how many bits are used to store one second of compressed audio. In general, this bitrate can either be defined as a fixed *constant bitrate* (CBR) or dynamically adjusted to the actual complexity of an audio signal, in form of a *variable bitrate* (VBR). The main advantage of an encoding with a CBR is merely a predictable file size. In contrast, the VBR can be a more efficient compression as it adapts the bitrate of the output signal to the actual complexity of the input signal. We focus our evaluation on the more efficient VBR. Specifically, we use MP3 with the average target bitrates of 245 kbit/s and 70 kbit/s. These bitrates produces results corresponding to both a *high* quality level (i. e., the compressed signals are indistinguishable for a human listener) and *low* quality level (i. e., the signal is highly compressed).

4.2 Metrics

For assessing the quality of adversarial examples both in terms of efficacy as well as inconspicuousness, we use two standard measures.

4.2.1 Word Error Rate (WER)

The *Word Error Rate* (WER) is computed based on the Levensthein distance [50], which describes the *edit distance* to transform one text into another (i.e., transform the output text of the ASR system into the correct text).

We compute the minimal Levensthein distance $\mathcal{L}$ as the sum over all substituted words S, inserted words I, and deleted words D.

$$\text{WER} = 100 \cdot \frac{\mathcal{L}}{N} = 100 \cdot \frac{S+D+I}{N},$$

where N is the total number of words of the reference text. The smaller the WER, the fewer errors were made by the ASR system.

To evaluate the efficacy of adversarial examples, we measure the WER between the adversarial target transcription and the output of the ASR system. Thus, a *successful adversarial example* has a WER of 0%, i. e., fully matching the desired target description y'. Note that the WER can also reach values above 100 %, e. g., when many words are inserted. This can especially happen with unsuccessful adversarial examples, where mostly the original text is transcribed, which leads to many insertions.

4.2.2 Segmental Signal-to-Noise Ratio (SNRseg)

The WER can only measure the success of an adversarial example in fooling an ASR system. For a real attack, we are also interested in the (in-)conspicuousness of adversarial examples, i. e., the level of the added perturbations. For this purpose, we quantify the changes that an attacker applies to the audio signal. Specifically, we use the *Signal-to-Noise Ratio* (SNR) to measure the added perturbations. More precisely, we compute the *Segmental Signal-to-Noise Ratio* (SNRseg) [51, 52], which is a more accurate measure of perceived distortion than the SNR, when signals are aligned [52].

Table 1: **Recognition rate of the ASR system on benign input.** We report the performance for the unmodified system as well as when hardened with MP3. Best WER is highlighted in **bold** for each input set.

		Standard Model	Augmented Model High Qual.	 Low Qual.
Unmodified Input		**WER 5.79%**	WER 9.59%	WER 7.87%
MP3 Input	High Qual.	WER 19.16%	**WER 7.74%**	-
	Low Qual.	WER 25.16%	-	**WER 8.29%**

Given the original audio signal $x(t)$ and the adversarial perturbations $\sigma(t)$ defined over the sample index t, the SNRseg can be computed via

$$\text{SNRseg(dB)} = \frac{10}{K} \sum_{k=0}^{K-1} \log_{10} \frac{\sum_{t=Tk}^{Tk+T-1} x^2(t)}{\sum_{t=Tk}^{Tk+T-1} \sigma^2(t)},$$

with T being the number of samples in a segment and K the total number of segments. For our experiments, we set the segment length to 16ms, which corresponds to $T = 256$ samples for a 16 kHz sampling rate.

The *higher* the SNRseg the *less* noise has been added to the audio signal, hence an adversarial example is considered as less conspicuous for higher SNRseg values. Note that we use the SNRseg ratio only as a rough approximation for the perceived noise. We perform a listening test with humans for a realistic assessment and show that the results of the listening test correlate with the reported SNRseg (cf. Section 4.6).

4.3 Benign Performance

To verify that our augmented model retains its practical use, we investigate its performance on benign samples, i. e., non-malicious, unaltered speech inputs.

Experiment Setup. We apply MP3 compression to the WSJ corpus to remove inaudible ranges from the training data. We consider both a high and low quality level for the MP3 compression and train for each quality level a hardened model with the updated speech corpus. As reference, we also train a baseline model on the original, unmodified speech corpus. Given these models, we decode the evaluation set `eval92` of the WSJ speech corpus and measure the accuracy both for unmodified inputs as well as when pre-processing with MP3 has been applied to the data.

Table 2: **Word Error Rate (WER) and number of successful adversarial examples (AEs) for a static attacker.** We report the numbers for all computed adversarial examples for the unmodified system as well as when hardened with MP3.

	Standard Model	Augmented Model High Qual.	 Low Qual.
Successful AEs Target Text	86/100 WER 3.81%	0/100 WER 204.98%	0/100 WER 196.78%
Original Text	WER 96.85%	WER 19.92%	WER 23.01%

Table 3: **Word Error Rate (WER) and number of successful adversarial examples (AEs) for the static attack using non-speech content.** We report the numbers for all computed adversarial examples for the unmodified system as well as when hardened with MP3. As music and bird sounds do not (generally) contain any spoken content, we report only the WER to the target phrase.

	Standard Model	Augmented Model High Qual.	 Low Qual.
Music	62/100 WER 15.78%	0/100 WER 99.71%	0/100 WER 99.41%
Birds	96/100 WER 0.9%	0/100 WER 99.55%	0/100 WER 99.70%

Results. The results are presented in Table 1. Notably, we obtain the worst WER of 19.16% and 25.16% when decoding MP3 pre-processed inputs with the model trained on the standard corpus. This confirms our suspicion that models not augmented with our pre-processing utilize information inaudible for humans as additional guidance. One could argue that the baseline is not trained for MP3-pre-processed input, however, if we decode raw audio with the MP3-trained model, the WER drops only slightly by 3.80% (9.59%) and 2.08% (7.87%) compared to the unaltered model (5.79%). This reinforces our belief that we indeed obtained a model which more closely resembles the human auditory system.

4.4 Static Attacker

To evaluate the effectiveness of the proposed defense against known, state-of-the-art attacks, we first consider an attacker that computes adversarial examples for the hardened system but is *oblivious* to the added MP3-based pre-processing step. For this purpose, we use the attack method described by Schönherr et al. [19] as introduced in Section 2.3, which can construct targeted audio adversarial examples for the KALDI toolkit.

Experiment Setup. For the attack, we randomly select 100 samples from the evaluation set `eval92` of the WSJ speech corpus. Following the authors [19], we bound the number of backpropagation steps to 500. Finally, Schönherr et al.'s attack uses "psychoacoustic hiding" to hide adversarial perturbations within inaudible parts of the spectrum, i. e., they specifically hide their perturbation in the parts of the audio we remove via MP3 compression. To avoid overfitting to their attack vector, we relax any psychoacoustics constraints. This change allows an attacker to make arbitrary changes to the input and, consequently, we do not constrain the added perturbations. For the evaluation, we use the same three models (two hardended, one baseline) from Section 4.3.

Results. The results of our experiment are shown in Table 2. We calculate two different WERs, one with respect to the target transcription ("Target Text" in Table 2) and one with respect to the original phrase ("Original Text" in Table 2). When decoding the adversarial examples, we observe that the attack *completely* fails for the hardened models. We were unsuccessful in obtaining an adversarial example (i. e., any adversarial example with a WER of 0% for the target transcription) for any of the tested audio files. Similarly, we can observe a drastically higher WER which increased to 200.59 % and 203.66 % from 3.81 % in comparison to the baseline model.

At the same time, the WER with respect to the original transcription decreases. Hence, we conclude that the ASR augmentation is very effective against such a *static* attacker, while at the same time is able to recover parts of the original transcription.

In the following, we analyze additional properties of our approach. Specifically, we want to answer the following questions:

- *Compression Rate.* How is the attack affected by different compression rates of the MP3 encoder?
- *Recovering Original Transcriptions.* Can we (partially) recover the original transcription from pre-processed adversarial examples?
- *Non-Speech Audio Files.* Is the defense also effective for audio files with no spoken content?

4.4.1 Compression Ratio

So far, we only considered the pre-processsing for two quality levels. Next, we test whether the results of these two bounds are representative for different compression ratios as well. Therefore, we repeat the experiment from Section 4.4 with ten different compression levels. These levels ranges from a *transparent* compression producing high quality results with an average target bitrate of 245 kbit/s to a compression with an average target bitrate of 70 kbit/ that produces low quality results.

As can be seen in Figure 3, we obtain comparable results between different compression rates. On average, we obtain a WER of 198.67 % ± 4.19 % for the ten experiments. Thus,

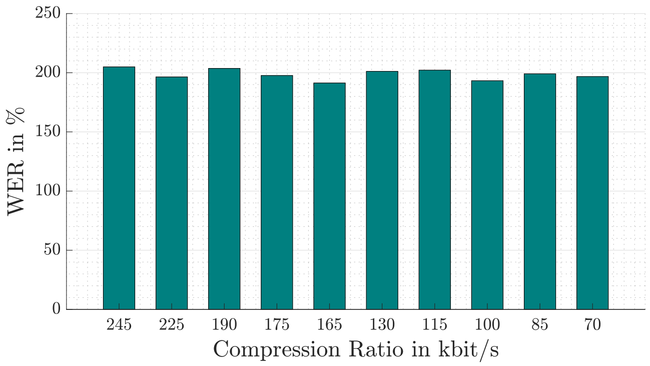


Figure 3: **Word Error Rate (WER) for the static attack against systems hardened with ten different levels of MP3 compression.**

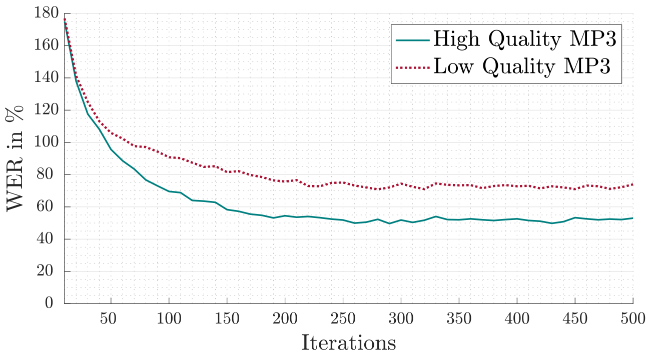


Figure 4: **Progress of the adaptive attack for computing adversarial examples.** We report the Word Error Rate (WER) for the first 500 iterations.

regardless of the compression rate, the attacker is not able to calculate successful adversarial examples.

4.4.2 Recovering Original Transcriptions

As discussed before, pre-processing adversarial examples computed by the static attacker with MP3 is an effective measure to eliminate most of the introduced perturbations. Rather than just destroying the malicious perturbations, the compression also allows the augmented ASR system to partially recover the original transcription.

To evaluate this effect, we further examine the adversarial examples from Section 4.4 and calculate the distance between the recognized hypothesis and the original transcription. As one can see in Table 2, before pre-processing the adversarial examples have a WER of 96.85% with respect to the original transcription. After pre-processing, this WER is significantly reduced (to 19.92% and 23.01%) and, thus, allows us to recover most of the original transcription.

The following example compares the quality of a transcription with different WERs:

```
WER   0.00%: I  AM  A   SPACE  INVADER   COMING   FOR  YOU
WER  12.50%: I  AM  A   SPACE  INVADER   COMING   FOR
WER  25.00%: I  AM  A   SPACE     IN     VOTING   FOR  YOU
WER  50.00%: I  AM  A   SPILL  INSIDER   COMING        TRUE
WER  75.00%: I  AM  HE  WILL   TRY    TO FIGHTING FOR
WER 100.00%:    NOW THE IS     TRYING TO MAKE      A    MAN
```

Hence, with a WER of ~20%, we can still recover most of the original transcription. While with a higher WER the quality gradually degrades.

4.4.3 Non-Speech Audio Content

Normally, the task of an ASR system is to transcribe audio files with spoken content. An attacker, however, might pick other content, i.e., music or ambient noise. In this case, it is still possible to create valid adversarial examples [19]. With an additional experiment, we want to analyze the properties of our defense when running the attack against such non-speech audio files. For this purpose, we use 100 samples of music and bird twittering and compute adversarial examples analogously to Section 4.4.

Similarly to the experiments with speech samples, the results in Table 3 show that the defense remains effective – independent of the input. In both cases, the attacker is not able to create valid adversarial examples and the WER increases to 99.41% and 99.70%. Note that the WER is smaller compared to the experiments with speech as the samples do not contain any spoken content, which would be transcribed by the ASR system.

4.5 Adaptive Attacker

In the previous section, we demonstrated how we can successfully thwart a static attack that does not *adapt* to our defense. With MP3-processing, we introduce a non-differentiable part into the ASR system and therefore it is not possible to calculate the gradient required for the attack.

However, as already demonstrated for the image domain [53], defenses against adversarial examples that rely solely on the *obfuscation* of gradients can typically reliably be circumvented. In fact, it has been shown by Carlini and Wagner [41] that the same techniques can also be applied in the audio domain.

Based on these insights, we now discuss how we can construct a stronger, more realistic, attack that can successfully create adversarial examples for our augmented ASR system. Given this new attack, we confirm that the adversarial examples are forced into human-perceptible ranges and, consequently, loose much of they malicious impact. We provide further evidence for this claim in Section 4.6 by performing a user study measuring the perceived quality of these examples.

Table 4: **Number of successful adversarial examples (AEs), Word Error Rate (WER) and Segmental Signal-to-Noise Ratio (SNRseg) for the experiment with the adaptive attacker.** We report the numbers for all computed adversarial examples against the augmented models as well as the two baselines with and without enabled *hearing thresholds* against the standard system. For the SNRseg we only consider successful AEs.

	Static Attack w/o thresh.	Static Attack w/ thresh.	Adaptive Attack	
Model	Standard	Standard	Augmented (High Qual.)	Augmented (Low Qual.)
Successful AEs Target Text	86/100 WER 3.81%	8/100 WER 76.28%	19/100 WER 50.81%	5/100 WER 68.37%
SNRseg	8.75±4.26 dB	17.86±1.94 dB	-10.03±3.23 dB	-12.12±1.20 dB

4.5.1 Straight-Through Estimator

We take inspiration from previous work [53] and use the so-called straight-through estimator [54] to strengthen the attack. Given the system $ASR(x) = y$, we can describe the hardened system as

$$ASR(MP3(x)) = y$$

Under the (crude) assumption that $x \approx MP3(x)$, we approximate the gradient for the attack as

$$\nabla_x ASR(MP3(x)) \approx \nabla_{\tilde{x}} ASR(\tilde{x})$$

with $\tilde{x} = MP3(x)$.

Specifically, we extend the static attack from Section 4.4 in two ways: first, for the forward propagation, the input is pre-processed with MP3. Second, for the backpropagation, we approximate the partial derivative of the MP3 function with the identity function.

Experiment. We compute adversarial examples for two hardened systems considering, again, a high and low quality level for the MP3-pre-processing. Since we want to analyze the (in-)conspicuousness of adversarial examples computed with the adaptive attack, we consider two baselines. First, we repeat the static attack from Section 4.4 against the unaltered system. Second, we enable the *hearing thresholds* for this attack (cf. 2.3), which restrict the amount of changes the attacker is allowed to make within audible time-frequency ranges.

For all four configurations, we compute adversarial examples for the speech dataset from Section 4.4 and run the attack for a maximum of 500 iterations.

Results. The results of our experiment are shown in Table 4. The adaptive attack is indeed successful in creating adversarial examples for the hardened system. Figure 4 shows the progress during the attack for both compression levels. Although successful for a couple of examples (19/100 and 5/100), the attack converges for both augmented systems at a WER of ~50% and ~70%, respectively. This indicates that the approximation of the gradient is not sufficient to generate adversarial examples in general, but still allows us to study the properties of *MP3-robust* adversarial examples. Also, suggested by the higher WER and smaller number of successful examples, we can increase the robustness of the system with an increased compression rate ("Low Qual." in Table 4).

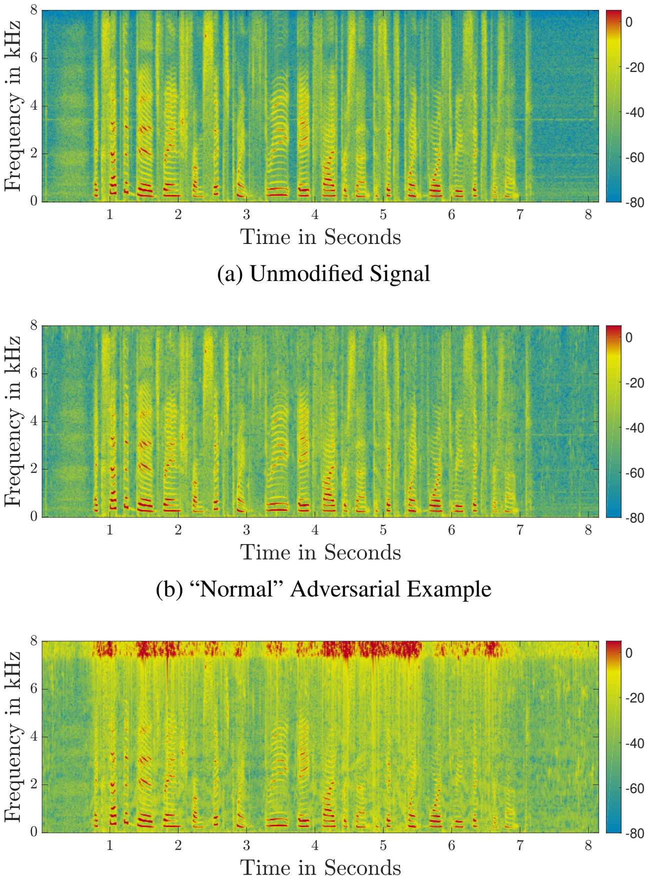
Considering the successful adversarial examples, we note that MP3-robust adversarial examples have a much lower and, importantly, negative SNRseg. This means that the energy of the noise (i.e., adversarial perturbations) exceeds the energy of the signal. Compared to the baselines, the noise energy increases on average by 20.87 dB (without hearing thresholds) and 29.98 dB (with hearing thresholds). Whereas, for the non-robust adversarial example, the original signal energy is about ten times higher than the energy of the adversarial perturbation. Thus, the situation is now reversed: there is ten times more energy in the perturbations than in the original audio signal. This is also evident from the power spectrum as depicted in Figure 5. Compared to the adversarial perturbations added by the static attacker (Figure 5b), the adaptive attacker (Figure5c) needs to make much more substantial changes of the original signal to overcome the augmented ASR system (Figure 5a).

4.6 Listening Tests

So far, the SNRseg provides an approximation of the amount of perturbations added for the attack. To study the actual perceived audio quality of generated adversarial examples, we have conducted a *Multiple Stimuli with Hidden Reference and Anchor* (MUSHRA) test [55]. This is a commonly used type of listening test to assess the quality of audio signals, which enables us to understand the practical impact of our defense in more detail.

4.6.1 Study Design

In a MUSHRA test, the participants are asked to rate the quality of a signal in different conditions from 0 (bad) to 100 (excellent). For our purpose, we want to rank the perceived quality of adversarial examples computed from the



(a) Unmodified Signal

(b) “Normal” Adversarial Example

(c) MP3 Robust Adversarial Example

Figure 5: **Spectrograms of adversarial examples computed with the static and adaptive attack.** (a) shows the unmodified signal. (b) shows an adversarial example computed with the static attack and enabled hearing thresholds, and (c) shows an adversarial example computed with the adaptive attack for a system augmented with a low quality level for the MP3-pre-processing.

adaptive attacker against the static attack. Therefore, we use the adaptive attack to compute MP3-robust adversarial examples against two systems augmented with a high and low quality level for the MP3 pre-processing. For the static attack, we calculate adversarial examples against the unaltered system and enable *hearing thresholds* for this attack (cf. 2.3) as we are interested in the inconspicuousness of the adversarialexamples.

In general, the test listeners are asked to rate a set of differently processed audio files. In our case, we ask for ratings of the computed adversarial examples. In addition, a MUSHRA test set contains a hidden *reference* and a so-called *anchor*. These are used to evaluate whether the participants are able to distinguish between the different audio conditions. The *reference* is a version with the best possible quality and we use the original audio file for this purpose. The *anchor* should be processed to have the worst quality across the entire set. For a given set, we construct this anchor as follows: We sum the noise of each of the three adversarial examples in the set and add this sum to the original signal, such that 1) each noise signal contributes the same amount of energy and 2) the SNR of the anchor is 3dB worse than the worst adversarial example in the set.

We have prepared a MUSHRA test with six of these sets based on different audio samples: two speech samples, two music samples, and two samples with bird song. The target text remained the same for all adversarial examples, and in all cases, the attacks were successful within 500 iterations, with the one exception of the music samples. In this case, we were not able to obtain successful adversarial examples and ran the attack against a system augmented with a slightly lower compression, at a bitrate of 85 kbit/s (instead of 70kbit/s).

4.6.2 Results

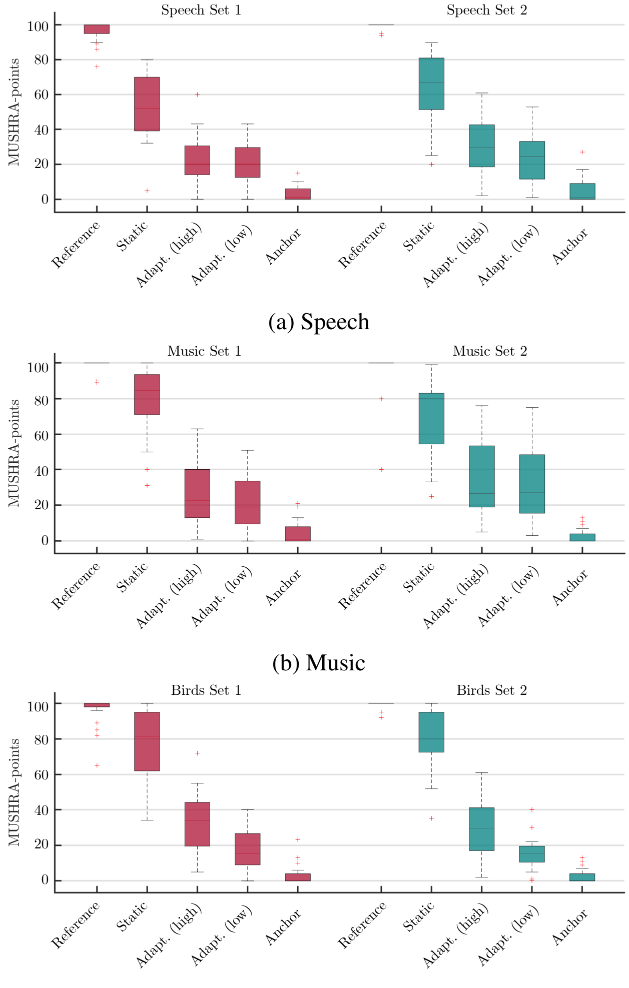
For the study, 20 participants completed the MUSHRA test. All participants were informed of the procedures and gave their consent. We used the *webMUSHRA* framework, which was developed by AudioLabs [56]. The listening test was conducted via headphones in a soundproofed chamber to minimize potential interfering noise. The results are presented in Figure 6.

Throughout the test sets, the participants successfully identified both the reference as the best and the anchor as the worst signal, which was confirmed by one-sided t-tests, where we tested whether the *anchor* was rated poorer than the MP3-hardened adversarial examples and the *reference* was rated better than the adversarial examples of the unaltered system with a significance level of 1 %.

For all test sets, the quality of the adversarial examples computed with the static attack was rated much better in comparison to the MP3-robust adversarial examples. These were unanimously rated with a poorer quality. This was also confirmed by one-sided t-tests with a significance level of 1 %. Furthermore, these results are consistent between different types of audio and are merely displaced on the y-Axis.

Between the MP3-robust adversarial examples we measured a significant difference for 3 out of the 6 sets. Nevertheless, both adaptive attacks exhibit notably poorer quality across all six evaluated sets. Further proving the effectiveness of the augmented ASR.

In conclusion, the results empirically support our hypothesis that augmenting the recognition system via MP3 compression forces the attacker to move the adversarial perturbations into audible ranges, which effectively makes the attack clearly noticeable for human listeners.



(a) Speech

(b) Music

(c) Birds

Figure 6: **Ratings of all 20 participants in the MUSHRA test.** In the user study, we tested six audio samples, divided into two samples each of spoken content, music and bird twittering.

5 Related Work

In this section, we summarize research related to our work, surveying recent attacks and countermeasures.

5.1 Audio Adversarial Examples

Carlini and Wagner [41] introduced targeted audio adversarial examples for ASR systems. For the attack, they assume a white-box attacker and use an optimization-based method to construct general adversarial examples for arbitrary target phrases against the ASR system DEEPSPEECH [34].

Similar, Schönherr et al. [19] and Yuan et al. have proposed an attack against the KALDI [37] toolkit. Both methods assume a white-box attacker and also use optimization-based methods to find adversarial examples. Furthermore, the attack from Schönherr et al. [19] can optionally compute adversarial examples that are especially unobtrusive for human listeners.

Neekhara et al. recently published a work in which they showed that it is possible to create so-called universal adversarial perturbations, which can be added to any audio and will lead to the wrong transcription [57]. Another work on universal adversarial perturbations for audio has been published by Abdoli et al., where the author were also successful to find universal perturbations for targeted attacks [58].

Alzantot proposed a black-box attack, which does not require knowledge about the model [59]. For this the authors have used a genetic algorithm to create their adversarial examples for a keyword spotting system. Khare et al. proposed a black-box attacks based on evolutionary optimization [60] and also Taori et al. presented a similar approach in their paper [61].

5.2 Countermeasures

There is a long line of research about countermeasures against adversarial examples in general and especially in the image domain (e. g., [24–26]), but most of the proposed defenses were shown to be broken once an attacker is aware of the employed mechanism. In fact, due to the difficulty to create robust adversarial example defenses, Carlini et al. proposed a guideline for the evaluation of adversarial robustness. They list all important properties of a successful countermeasure against adversarial examples [32].

Compared to the image domain, defenses against audio adversarial examples remained relatively unnoticed so far. For the audio domain, only a few works have investigated possible countermeasures. Moreover, these tend to focus on specific attacks and not adaptive attackers.

Akinwande et al. [62] and Samizade et al. [63] proposed to detect adversarial examples by searching for anomalies and artifacts either in the networks activation's or directly on raw audio. Similar, Ma et al. [64] describe how the correlation of audio and video streams can be used to detect adversarial examples for the Audiovisual Speech Recognition task. However, all of these simple approaches—while reasonable in principle—are specifically trained for a defined set of attacks and hence an attacker can easily leverage that knowledge as demonstrated repeatedly in the image domain [26].

Zeng et al. [65] proposed an approach inspired by multiversion programming. Therefore, the authors combine, the output of multiple ASR systems and calculate a *similarity score* between the transcriptions. If these differ two much, the input is assumed to be an adversarial example. The security of this approach relies on the property that current audio adversarial examples do not transfer between systems. An assumption which has been already shown to be wrong in the image domain [66].

Rajaratnam et al. [67] use the observation that ASR systems are typically relatively robust against natural noise and calculate for each input a so-called *flood score*. This score indicates how much noise can be added until the hypothesis of the system changes. Given that adversarial perturbations are fragile, this score can then be used for detection. Although, shown to be effective against static attackers, this approach is, again, not resistant against adaptive attackers who can simply integrate this flooding into the attack once the details are available.

Yang et al. [68], also utilize specific properties of the audio domain and use the temporal dependency of the input signal. For this, they compare the transcription of the whole utterance with a segment-wise transcription of the utterance. In case of a benign example, both transcriptions should be the same, which will not be the case for a adversarial example. This proofed effective against static attacks and the authors also construct and discussed various adaptive attacks.

In contrast, our approach does not rely on detection by augmenting the entire system to become more resilient against adversarial examples. The basic principle of this has been discussed as a defense mechanism in the image domain with JPEG compression [69, 70] as well as in the audio domain by Carlini and Wagner [41] and Rajaratnam et al. [71]. However, in all of theses cases, it was solely used as a pre-processing step in order to destroy adversarial perturbations rather than confine the nature of the perturbations itself.

6 Discussion

We have shown how we can augment an ASR system with psychoacoustic compression. In particular, we use the MP3 codec to effectively remove semantically irrelevant information from audio signals. This allowed us to train a hardened model that is more similar to human perception.

Model Hardening Our results from Section 4.3 suggest that the hardened models indeed primarily utilize information available within audible ranges. Specifically, we observe that models trained on the unmodified dataset appear to use all available signals and utilize information *both* from audible and non-audible ranges. This is reflected in the accuracy drop when presented with MP3 pre-processed input (where only audible ranges are available). In contrast, the hardened model performs only slightly worse when presented with raw audio. Here, the model focuses on the available audible ranges and *ignores* all inaudible parts. Of course, the loss of accuracy can also be caused (at least partially) by using input data types for which the system has not been trained. However, this should also affect the augmented model when decoding raw audio.

Robustness of the System As demonstrated in the experiments for the static attacker, MP3 pre-processing is very effective in destroying adversarial perturbation added by an attacker *oblivious* of the defense. This is further supported by the fact that we can recover most of the *original* transcription.

We demonstrated how we can create a more realistic attacker, which actively avoids the defense. In this case, however, the attack is forced into the audible range. This makes the attack significant more perceptible — resulting in a SNRseg drop of 29.98 dB on average to values much smaller than 0, where therefore the adversarial perturbations contain far more energy—up to ten times—than the original audio signal. These results are further confirmed by the listening test conducted in Section 4.6, which emphasizes that the attack is clearly perceivable and that the adversarial examples, calculated with the adaptive attack, are easily distinguishable from benign audio files.

Parameter Choice In general, the defense can be implemented as low-cost pre-processing step with no noteworthy performance overhead. In order to deploy the defense, we need to re-train the system with the updated speech corpus to achieve an accuracy within a similar range as the baseline system.

The results from the attacks with the adaptive attacker also suggest that we gain a stronger defense when using MP3 to compress inputs to a lower quality. Specifically, for our experiments the attacker was only able to compute 5 successful adversarial examples out of 100 samples in this case. Also, the noise energy increased on average by an additional 2dB compared to a lower compression ratio.

Improvement of the Attack The adaptive attack presented in Section 4.5 is able to successfully compute MP3-robust adversarial examples. However, the results in Figure 4, in which we have plotted the WER in relation to the number of iterations of the attack, indicate that most of the adversarial examples can not be further improved. These will most likely not lead to the target transcription, even if the attack runs for more iterations.

This is not surprising, as we use an approximation of the MP3 pre-processing gradient. The obvious solution to this issue is to increase the information available to the attacker and integrate MP3 into the backpropagation. However, MP3 is a non-differentiable function and therefore can not be directly considered in the optimization criterion. In this work, we focus on obtaining a closer similarity to the human auditory system within an ASR and on determining, which effect this has on adversarial examples. Additionally, we experimented with different approximations (other than the one presented in Section 4.5) of MP3, but could not find a more suitable one. In any case, the attack is forced into audible ranges and consequently, generates adversarial perturbations, perceivable for human listeners.

A further improvement could be to reduce the adversarial perturbations from the attack with an additional MP3 *post*-processing step. In this case, we might be able to remove noise that would be deleted anyway during the pre-processing of the augmented system. We tried this and applied MP3 compression before feeding it into the hardened ASR system (which, by design, applies MP3 compression again). This, however, failed, as the calculated hearing thresholds depend on the input signal, which change after applying MP3 to the signal.

Future Work In this work we have shown how we can force adversarial audio attacks into audible ranges, which makes them clearly perceivable. Ultimately, the goal is to push adversarial examples towards the perceptual boundary between original and adversarial message. Intuitively, adversarial examples should require such extensive modification that a *human listener* will perceive the target transcription, i. e., that the adversarial perturbation carries *semantic* meaning. We leave the exploration of this strategy as an interesting question for future work.

7 Conclusion

In this work, we presented a countermeasure for audio adversarial examples. To achieve this, we have shown that the augmentation of an ASR system with MP3-compression can effectively prevent adversarial examples. While at the same time, the performance on benign audio remains robust. Effectively, the system leveraged psychoacoustic hearing thresholds to use only parts of the audio perceivable by humans.

We have argued that for any kind of countermeasure, an attacker will find adversarial examples, especially, if we assume the attack to have full access to the system. However, we have demonstrated that the perturbations of the original signal can be forced to be so high that they become obvious for humans. We have calculated adversarial examples for our proposed system via an adaptive attack, which specifically leverages the knowledge of the proposed countermeasure. Although some successful adversarial examples have been obtained by the adaptive attacker, the attack in general is much less effective in sense of the calculated WER with respect to the target transcription, as well as the measured SNRseg. For the SNRseg we measured an average value smaller than 0, showing that the energy of the noise required to calculate successful adversarial examples is, up to ten times, higher than the original signal's energy itself.

We additionally confirm that the noise is clearly perceptible and significantly impairs the audio signal's quality, via a listening test. During this listening test we compared the original attack with the adaptive one, demonstrating that the original attack was rated with a much higher quality than the adversarial examples of the adaptive attack.

In general we have taken the first steps towards bridging the gap between human expectations and the reality of ASR systems.

References

[1] Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web*. 2007.

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[3] David Silver Andrei A. Rusu Joel Veness Marc G. Bellemare Alex Graves Martin Riedmiller Andreas K. Fidjeland Georg Ostrovski Stig Petersen Charles Beattie Amir Sadik Ioannis Antonoglou Helen King Dharshan Kumaran Daan Wierstra Shane Legg Demis Hassabis Volodymyr Mnih, Koray Kavukcuoglu. Human-level control through deep reinforcement learning. *Nature*, 2015.

[4] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 2016.

[5] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

[6] Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 2019.

[7] Andrew W Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander WR Nelson, Alex Bridgland, et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 2020.

[8] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[9] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. Achieving human parity in conversational speech recognition. *CoRR*, 2016.

[10] Ashwin Ram, Rohit Prasad, Chandra Khatri, Anu Venkatesh, Raefer Gabriel, Qing Liu, Jeff Nunn, Behnam Hedayatnia, Ming Cheng, Ashish Nagar, et al. Conversational ai: The science behind the alexa prize. *Advances*

in Neural Information Processing Systems (NeurIPS) - (Conversational AI Workshop), 2017.

[11] Laren Goode. Amazon's alexa will now lock your door for you (if you have a 'smart' lock). https://www.theverge.com/circuitbreaker/2016/7/28/12305678/amazon-alexa-works-with-august-smart-lock-door-WiFi-bridge.

[12] Stephen Shankland. Meet Tesla's self-driving car computer and its two AI brains. https://www.cnet.com/news/meet-tesla-self-driving-car-computer-and-its-two-ai-brains/.

[13] 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.

[14] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. *Lecture Notes in Computer Science*, page 387–402, 2013.

[15] Nicolas Papernot, Patrick D. McDaniel, and Ian J. Goodfellow. Transferability in machine learning: From phenomena to Black-Box attacks using adversarial samples. *CoRR*, abs/1605.07277:1–13, May 2016.

[16] Liwei Song and Prateek Mittal. Poster: Inaudible voice commands. In *ACM Conference on Computer and Communications Security (CCS)*, 2017.

[17] Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. DolphinAttack: Inaudible voice commands. In *Conference on Computer and Communications Security*, pages 103–117. ACM, October 2017.

[18] Xuejing Yuan, Yuxuan Chen, Yue Zhao, Yunhui Long, Xiaokang Liu, Kai Chen, Shengzhi Zhang, Heqing Huang, Xiaofeng Wang, and Carl A. Gunter. CommanderSong: A systematic approach for practical adversarial voice recognition. *arXiv preprint arXiv:1801.08535*, 2018.

[19] Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. In *Network and Distributed System Security Symposium (NDSS)*, 2019.

[20] Lea Schönherr, Thorsten Eisenhofer, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. Imperio: Robust over-the-air adversarial examples for automatic speech recognition systems. In *arXiv preprint arXiv:1908.01551*.

[21] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings, 2015.

[22] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks, 2015.

[23] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *Symposium on Security and Privacy*, pages 39–57. IEEE, May 2017.

[24] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations, 2017.

[25] Reuben Feinman, Ryan R. Curtin, Saurabh Shintre, and Andrew B. Gardner. Detecting adversarial samples from artifacts, 2017.

[26] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *ACM Workshop on Artificial Intelligence and Security*, 2017.

[27] Justin Gilmer, Ryan P Adams, Ian Goodfellow, David Andersen, and George E Dahl. Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*, 2018.

[28] Adi Shamir, Itay Safran, Eyal Ronen, and Orr Dunkelman. A simple explanation for the existence of adversarial examples with small hamming distance. *arXiv preprint arXiv:1901.10861*, 2019.

[29] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[30] Eberhard Zwicker and Hugo Fastl. *Psychoacoustics: Facts and Models*. Springer, third edition, 2007.

[31] ISO. Information Technology – Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbits/s – Part3: Audio. ISO 11172-3, International Organization for Standardization, 1993.

[32] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, and Aleksander Madry. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019.

[33] Herve A Bourlard and Nelson Morgan. *Connectionist speech recognition: a hybrid approach*, volume 247. Springer Science & Business Media, 2012.

[34] 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.

[35] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*, pages 1764–1772, 2014.

[36] Jian Kang, Wei-Qiang Zhang, Wei-Wei Liu, Jia Liu, and Michael T. Johnson. Advanced recurrent network-based hybrid acoustic models for low resource speech recognition. *EURASIP Journal on Audio, Speech, and Music Processing*, 2018(1):6, Jul 2018.

[37] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely. The kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, December 2011. IEEE Catalog No.: CFP11SRW-USB.

[38] Yao Qin, Nicholas Carlini, Ian Goodfellow, Garrison Cottrell, and Colin Raffel. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *arXiv preprint arXiv:1903.10346*, March 2019.

[39] Hadi Abdullah, Washington Garcia, Christian Peeters, Patrick Traynor, Kevin R. B. Butler, and Joseph Wilson. Practical hidden voice attacks against speech and speaker recognition systems. In *Network and Distributed System Security Symposium (NDSS)*, 2019.

[40] Joseph Szurley and J Zico Kolter. Perceptual based adversarial audio attacks. *arXiv preprint arXiv:1906.06355*, 2019.

[41] Nicholas Carlini and David Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *IEEE Security and Privacy Workshops (SPW)*, 2018.

[42] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. *CoRR*, abs/1804.08598:1–10, April 2018.

[43] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction APIs. In *USENIX Security Symposium*, pages 601–618. USENIX, August 2016.

[44] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical Black-Box attacks against machine learning. In *Asia Conference on Computer and Communications Security (ASIA CCS)*, pages 506–519. ACM, April 2017.

[45] Binghui Wang and Neil Zhenqiang Gong. Stealing hyperparameters in machine learning. In *Symposium on Security and Privacy*. IEEE, May 2018.

[46] Auguste Kerckhoffs. La cryptographic militaire. *Journal des sciences militaires*, 1883.

[47] 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.

[48] Mike Cheng, Mark Taylor, Takehiro Tominaga, Naoki Shibata, Frank Klemm, Gabriel Bouvigne, Alexander Leidinger, Roberto Hegemann, and Rogerio Brito. The lame project. http://lame.sourceforge.net.

[49] Douglas B Paul and Janet M Baker. The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, pages 357–362. Association for Computational Linguistics, 1992.

[50] Gonzalo Navarro. A guided tour to approximate string matching. *ACM computing surveys (CSUR)*, 33(1):31–88, 2001.

[51] S Voranl and Connie Sholl. Perception-based objective estimators of speech. In *Proceedings. IEEE Workshop on Speech Coding for Telecommunications*, pages 13–14. IEEE, 1995.

[52] Wonho Yang. *Enhanced Modified Bark Spectral Distortion (EMBSD): An Objective Speech Quality Measure Based on Audible Distortion and Cognitive Model*. Temple University, 1999.

[53] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. 2018.

[54] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[55] Nadja Schinkel-Bielefeld, Netaya Lotze, and Frederik Nagel. Audio quality evaluation by experienced and inexperienced listeners. In *International Congress on Acoustics*, pages 6–16. ASA, June 2013.

[56] Michael Schoeffler, Sarah Bartoschek, Fabian-Robert Stöter, Marlene Roess, Susanne Westphal, Bernd Edler, and Jürgen Herre. webMUSHRA – A comprehensive framework for web-based listening tests. *Journal of Open Research Software*, 6(1), February 2018.

[57] Prakhar Pandey Shlomo Dubnov Julian McAuley Farinaz Koushanfar Paarth Neekhara, Shehzeen Hussain. Universal adversarial perturbations for speech recognition systems. *Proceedings of Interspeech*, 2019.

[58] Sajjad Abdoli, Luiz G Hafemann, Jérôme Rony, Ismail Ben Ayed, Patrick Cardinal, and Alessandro L. Koerich. Universal adversarial audio perturbations. *arXiv preprint arXiv:1908.03173v2*, 2019.

[59] Moustafa Alzantot, Bharathan Balaji, and Mani Srivastava. Did you hear that? adversarial examples against automatic speech recognition. *arXiv preprint arXiv:1801.00554*, 2018.

[60] Senthil Mani Shreya Khare, Rahul Aralikatte. Adversarial black-box attacks on automatic speech recognition systems using multi-objective evolutionary optimization. *Proceedings of Interspeech*, 2019.

[61] Rohan Taori, Amog Kamsetty, Brenton Chu, and Nikita Vemuri. Targeted adversarial examples for black box audio systems. *arXiv preprint arXiv:1805.07820*, 2018.

[62] Victor Akinwande, Celia Cintas, Skyler Speakman, and Srihari Sridharan. Identifying audio adversarial examples via anomalous pattern detection, 2020.

[63] Saeid Samizade, Zheng-Hua Tan, Chao Shen, and Xiaohong Guan. Adversarial example detection by classification for deep speech recognition, 2019.

[64] Pingchuan Ma, Stavros Petridis, and Maja Pantic. Detecting adversarial attacks on audio-visual speech recognition, 2019.

[65] Qiang Zeng, Jianhai Su, Chenglong Fu, Golam Kayas, and Lannan Luo. A multiversion programming inspired approach to detecting audio adversarial examples, 2018.

[66] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples, 2016.

[67] Krishan Rajaratnam and Jugal Kalita. Noise flooding for detecting audio adversarial examples against automatic speech recognition. *2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Dec 2018.

[68] Zhuolin Yang, Bo Li, Pin-Yu Chen, and Dawn Song. Characterizing audio adversarial examples using temporal dependency. *arXiv preprint arXiv:1809.10875*, 2018.

[69] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M Roy. A study of the effect of jpg compression on adversarial images. *arXiv preprint arXiv:1608.00853*, 2016.

[70] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Siwei Li, Li Chen, Michael E Kounavis, and Duen Horng Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

[71] Krishan Rajaratnam, Kunal Shah, and Jugal Kalita. Isolated and ensemble audio preprocessing methods for detecting adversarial examples against automatic speech recognition, 2018.