Zero-Query Adversarial Attack on Black-box Automatic Speech Recognition Systems

Zheng Fang*
Wuhan University
Wuhan, China
zhengfang618@whu.edu.cn

Tao Wang*
Wuhan University
Wuhan, China
WTBantoeC@whu.edu.cn

Lingchen Zhao*†
Wuhan University
Wuhan, China
lczhaocs@whu.edu.cn

Shenyi Zhang*
Wuhan University
Wuhan, China
shenyizhang@whu.edu.cn

Bowen Li*
Wuhan University
Wuhan, China
bowenli0427@whu.edu.cn

Yunjie Ge*
Wuhan University
Wuhan, China
yunjiege@whu.edu.cn

Qi Li‡
Tsinghua University
Beijing, China
qli01@tsinghua.edu.cn

Chao Shen§
Xi'an Jiaotong University
Xi'an, China
chaoshen@mail.xjtu.edu.cn

Qian Wang*
Wuhan University
Wuhan, China
qianwang@whu.edu.cn

Abstract

In recent years, extensive research has been conducted on the vulnerability of ASR systems, revealing that black-box adversarial example attacks pose significant threats to real-world ASR systems. However, most existing black-box attacks rely on queries to the target ASRs, which is impractical when queries are not permitted. In this paper, we propose ZQ-Attack, a transfer-based adversarial attack on ASR systems in the zero-query black-box setting. Through a comprehensive review and categorization of modern ASR technologies, we first meticulously select surrogate ASRs of diverse types to generate adversarial examples. Following this, ZQ-Attack initializes the adversarial perturbation with a scaled target command audio, rendering it relatively imperceptible while maintaining effectiveness. Subsequently, to achieve high transferability of adversarial perturbations, we propose a sequential ensemble optimization algorithm, which iteratively optimizes the adversarial perturbation on each surrogate model, leveraging collaborative information from other models. We conduct extensive experiments to evaluate ZQ-Attack. In the over-the-line setting, ZQ-Attack achieves a 100% success rate of attack (SRoA) with an average signal-to-noise ratio (SNR) of 21.91dB on 4 online speech recognition services, and attains an average SRoA of 100% and SNR of 19.67dB on 16 open-source ASRs. In the over-the-air setting, ZQ-Attack also achieves a 100% SRoA with an average SNR of 15.77dB on 2 commercial intelligent voice control devices.

CCS Concepts

• **Security and privacy → Software and application security**;
• **Computing methodologies → Artificial intelligence**.

Keywords

Speech recognition; adversarial example attacks

*Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University
†Lingchen Zhao is the corresponding author.
‡Institude for Network Sciences and Cyberspace & BNRist
§School of Cyber Science and Engineering

The full version of this paper is available at [16].

1 Introduction

Automatic speech recognition (ASR) techniques, which convert spoken language into text, play a crucial role in modern human-computer interactions. These techniques are now widely employed in online speech recognition services [40, 42] provided by major companies, including Microsoft, OpenAI, etc. Online repositories also offer a plethora of open-source ASRs for public utilization. Furthermore, ASRs have also been integrated into commercial intelligent voice control (IVC) devices, such as Apple Siri [7] and Amazon Alexa [5], enabling users to perform various tasks via voice commands. Unfortunately, similar to other deep neural networks (DNNs) based systems, modern ASRs are vulnerable to adversarial attacks [10, 18, 55]. Attackers can introduce small perturbations into audio samples, causing the target ASR system to produce incorrect results.

Audio Adversarial Attacks. Recently, numerous studies have investigated the practicality and effectiveness of audio adversarial

Table 1: Summary of existing audio adversarial attacks.

Method	Setting	Knowledge	Target ASR [*]	Over-the-line [†]	Over-the-air	Queries [‡]
Carlini *et al.* [10]	White-box	Gradient	□	100%	-	~1000
Commandersong [64]	White-box	Gradient	□	100%	~80%	~1000
Taori *et al.* [56]	Black-box	Prediction score	□	~40%	-	~300,000
SGEA [59]	Black-box	Prediction score	□	100%	-	~100,000
Devil's Whisper [11]	Black-box	Confidence score	△★	~60%	~50%	~1500
Occam [68]	Black-box	Final decision	□△	100%	-	~10,000
KENKU [61]	Black-box	None	△★	~70%	~80%	>0
NI-Occam [68]	Zero-query black-box [§]	None	★	-	~50%	0
TransAudio [47]	Zero-query black-box	None	□△	~30%	-	0
ZQ-Attack	Zero-query black-box	None	□△★	100%	100%	0

Note that, (i) [*]: We use □, △, ★ to represent open-source ASRs, online speech recognition services, and commercial IVC devices, respectively. (ii) [†]: In the over-the-line and over-the-air settings, we employ '-' to represent unsuccessful attacks. In cases of successful attacks, the success rate of attack is presented. (iii) [‡]: "Queries" refer to the number of queries made to the target ASR system during the generation process. (iv) [§]: In contrast to the black-box setting, the zero-query black-box setting prohibits any queries to the target ASR system during the generation process.

attacks on ASR systems, as summarized in Table 1. Depending on the accessibility of the attacker to the target ASR systems, audio adversarial attacks can be categorized into white-box and black-box attacks. The initial works [10, 64] employ gradient descent algorithms to generate adversarial audios in the white-box setting, where attackers have full access to the internal information of the target ASR systems. However, in real-world scenarios, attackers typically lack access to the internal information of the target system, making these white-box attacks often impractical. To achieve attacks in black-box settings, several methods have been proposed to generate adversarial examples based on the limited information acquired through queries to the target ASR system [11, 56, 61, 68]. However, these methods require huge financial costs and time investments for the queries to generate a single adversarial example, and the large number of highly similar queries makes these attacks easily detectable by the target ASR system, rendering them impractical.

Therefore, to further enhance the practicality of adversarial attacks, researchers have increasingly turned their attention to transfer-based attacks, which can generate adversarial examples effectively across different target systems, thereby eliminating the need for queries. However, the performance of existing transfer-based audio adversarial example attacks also exhibits significant limitations. For instance, NI-Occam [68] generates audio adversarial examples on fine-tuned Kaldi models to attack IVC devices, but its attack success rate is quite limited. TransAudio [47] optimizes adversarial examples on a surrogate ASR model and can successfully attack black-box ASR systems with similar architectures to the surrogate model. However, it only achieves word-level modifications to the original transcription and has a low success rate on online speech recognition services. These results indicate that current transfer-based attacks still possess limited transferability.

Consequently, we propose the following question: *How to generate audio adversarial examples with high transferability in the challenging zero-query black-box setting?*

ZQ-Attack. Our answer to this question is ZQ-Attack, a transfer-based adversarial attack on black-box ASR systems without the need for queries. Inspired by the ensemble method [9, 38, 44], our core idea is to optimize the adversarial perturbation on diverse surrogate ASRs. Ideally, adversarial perturbations optimized concurrently on multiple different surrogate models should contain features that can be captured by these models, making them effective against various ASR systems.

Specifically, ZQ-Attack consists of three stages: surrogate ASRs selection, perturbation initialization, and sequential ensemble optimization. For the surrogate ASRs selection stage, we conduct an extensive survey of modern ASR systems and observe that different types of ASR systems utilize distinct acoustic models. Therefore, we first need to select multiple different types of surrogate ASRs. Then, to ensure that the generated adversarial perturbations are effective across these surrogate ASRs while maintaining high stealthiness, we propose an adaptive search algorithm that uses scaled target commands to initialize the adversarial perturbation, instead of using zeros or Gaussian noise as in existing methods. Following the initialization, ZQ-Attack employs a sequential ensemble optimization algorithm to collaboratively optimize the adversarial perturbations on the sequence of diverse surrogate ASRs. This sequential ensemble algorithm allows for the optimization of adversarial perturbations on each surrogate ASR while concurrently leveraging information from prior surrogate ASRs. Consequently, the generated adversarial perturbations are effective not only for the current surrogate ASR but also for the previous ASRs in the sequence.

We conduct extensive experiments in both over-the-line and over-the-air settings to validate the effectiveness and imperceptibility of our ZQ-Attack. In the over-the-line setting, ZQ-Attack achieves an average success rate of attack (SRoA) of 100% and signal-to-noise ratio (SNR) of 21.91dB on four online speech recognition services. Additionally, ZQ-Attack attains an average SRoA of 100% and SNR of 19.67dB on 16 open-source ASRs. In the over-the-air setting, ZQ-Attack achieves an average SRoA of 100% and an SNR of 15.77dB on two commercial IVC devices. These results demonstrate that ZQ-Attack can successfully generate audio adversarial examples with high transferability, effectively targeting various ASR systems without requiring any queries.

Contributions. Our contributions are summarized as follows:

- To the best of our knowledge, we are the first to generate audio adversarial examples with high transferability on ASR systems in the most challenging zero-query black-box setting. Our method is effective in both over-the-line and over-the-air settings, with the target ASR systems encompassing online speech recognition services, open-source ASRs, and commercial IVC devices, showcasing remarkable practicality.
- We introduce ZQ-Attack, a zero-query, transfer-based adversarial attack on black-box ASR systems. This approach optimizes adversarial perturbations using a set of diverse surrogate ASRs simultaneously, thereby enhancing their transferability. Furthermore, we develop an adaptive adversarial perturbation initialization method based on the target command audio to improve the imperceptibility.
- We conduct comprehensive experiments to evaluate the performance of ZQ-Attack on online speech recognition services, open-source ASRs, and commercial IVC devices. The experimental results demonstrate the superior performance of our method. ZQ-Attack can successfully attack all the target ASR systems without queries while achieving an average SRoA of 100%.

2 Background

2.1 Automatic Speech Recognition

ASR systems can automatically transcribe input audio into the corresponding transcriptions. As shown in Figure 1, a typical ASR system comprises three components: pre-processing, acoustic model, and decoder.

- *Pre-processing*. Given an input audio with n sample points, denoted as $x \in \mathbb{Z}^n$, an ASR system first normalizes it to the range of $[-1, 1]^n$ and applies low/high-pass filters to remove frequencies and segments beyond the range of human hearing. Then, the ASR system employs time-frequency transformation techniques, such as the short-time Fourier transform (STFT), to convert the time-domain signal x into the frequency domain spectrogram $S \in \mathbb{R}^{T \times F}$, where T and F denote the number of frames and frequency bins, respectively. In the subsequent steps, traditional ASR systems utilize acoustic feature extraction algorithms, such as mel-frequency cepstral coefficients (MFCC) [54], linear predictive coefficient (LPC) [32], or perceptual linear predictive (PLP) [28], to further transform S into meticulously designed acoustic features. In contrast, modern DNN-based ASR systems directly utilize S.
- *Acoustic model*. The acoustic model is typically a machine learning model that maps the spectrogram or extracted acoustic features to an intermediate representation. Traditional ASR systems use hidden Markov models (HMMs) and Gaussian mixture models (GMMs) [17, 49], while modern ASR systems employ DNNs, such as convolutional neural networks (CNNs) and Transformers [23, 36].
- *Decoder*. The decoder uses the intermediate representation to predict tokens and generate corresponding transcriptions. A token is the smallest unit of the transcription, and the set of all possible tokens constitutes a vocabulary V. The vocabulary V varies across different ASR systems. For example, $V = \{a, b, \cdots, z, space\}$ is a simple vocabulary for an ASR system recognizing English.

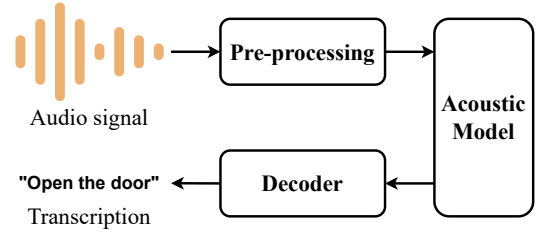**Figure 1: The architecture of a typical ASR system.**

2.2 Audio Adversarial Attacks

Audio adversarial attacks aim to manipulate the output of ASR systems using audio adversarial examples constructed by adding imperceptible perturbations to benign carrier audios [10, 11, 61, 64, 68]. Depending on the objective of the attack, audio adversarial attacks can be categorized into targeted and untargeted attacks.

Formally, let $f(x) : x \rightarrow y$ denote an ASR system that transcribes an audio x into the transcription $y = f(x)$, and let $x' = x + \delta$ denote the adversarial example constructed by adding the adversarial perturbation δ to the carrier audio x. An untargeted adversarial attack aims to mislead the target ASR system, causing it to produce any result other than the ground truth, represented as $f(x') \neq y$. In contrast, a targeted adversarial attack aims to induce the output of the target ASR system into a specific target transcription $t \neq f(x)$, which is formulated as:

$$f(x') = t, \quad s.t. \ Dis(x, x') < \epsilon, \tag{1}$$

where $Dis(x, x')$ represents the distance between x and x', commonly calculated using the L_p norm, with p usually being 0, 2, or ∞. ϵ is a hyper-parameter that constrains this distance. As targeted adversarial attacks can naturally extend to untargeted attacks, the adversarial attacks discussed in the rest of this paper will specifically refer to the targeted ones unless explicitly specified otherwise.

Typically, the generation process of adversarial examples can be formulated as an optimization problem:

$$\min_{\delta} \mathcal{L}(x, \delta, t, f) = \mathcal{L}_a(x, \delta, t, f) + c \cdot \mathcal{L}_p(\delta), \tag{2}$$

where the adversarial loss $\mathcal{L}_a$ measures the effectiveness of δ on the target ASR system f, and the imperceptibility loss $\mathcal{L}_p$ quantifies the imperceptibility of δ. The parameter c acts as a weighting factor, balancing the effectiveness and imperceptibility of the attack. Gradient descent is a common method for solving this optimization problem. It can be formulated as:

$$\delta \leftarrow clip_\epsilon \left(\delta - \alpha \cdot \nabla_\delta \mathcal{L}(x, \delta, t, f) \right), \tag{3}$$

where α represents the learning rate and $\nabla_\delta \mathcal{L}(x, \delta, t, f)$ is the gradient of $\mathcal{L}(x, \delta, t, f)$ with respect to δ. The function $clip_\epsilon$ limits $Dis(x, x')$ to a relatively small range controlled by ϵ.

3 Threat Model & Challenges

3.1 Threat Model

Goals. The attacker aims to generate audio adversarial examples that can be recognized as the target transcription by the target ASR systems without any queries. In the over-the-line setting (*i.e.,*
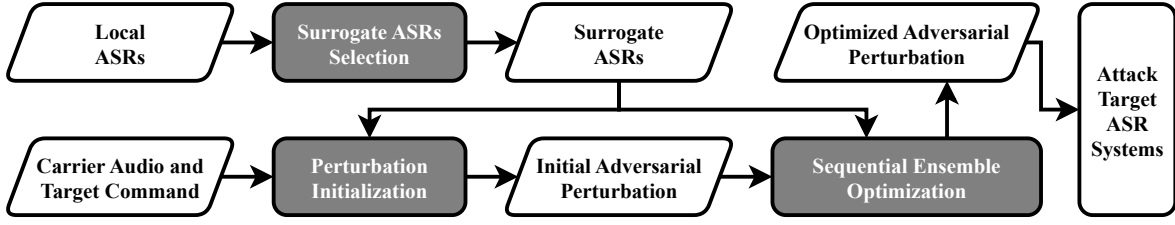
Figure 2: Workflow of ZQ-Attack. ZQ-Attack is mainly divided into three stages: surrogate ASRs selection, perturbation initialization, and sequential ensemble optimization.

digital attacks), the target systems are online speech recognition services or open-source ASRs. The waveform files of the audio adversarial examples are directly used as inputs, inducing the target ASR systems to produce the specified target transcriptions. In the over-the-air setting (*i.e.*, physical attacks), the target ASR systems are commercial IVC devices. The adversarial examples should be misrecognized as the target transcription by these devices after being transmitted through the air. Additionally, the audio adversarial examples should be imperceptible, making them difficult for human ears to detect.

Knowledge & Capabilities. Prior works on black-box adversarial attacks on ASR systems do not require the attacker to know internal information about the target ASR system, but they still assume that the attacker can interact with the target system. In this paper, we consider a more realistic and challenging scenario where the attacker also cannot query the target ASR system during the generation of adversarial examples. After generating the adversarial examples, the attacker can execute the attacks by uploading audio files to target ASR systems in the over-the-line setting and positioning a speaker near the target commercial IVC devices in the over-the-air setting. Additionally, we assume the attacker has the capability to train surrogate ASRs or obtain pre-trained surrogate ASRs from open-source repositories.

3.2 Challenges

To launch audio adversarial attacks in the challenging zero-query black-box setting, an alternative approach is the transfer-based attack. The basic idea of transfer-based attacks is to use a local surrogate model to generate adversarial examples that are then used to attack the target black-box model. Despite the demonstrated effectiveness of transfer-based attacks in the image domain [12, 38, 45, 60], achieving similar success in the audio domain remains an unresolved issue. Abdullah *et al.* [3] reveal that the transferability of audio adversarial examples among different ASR systems is exceedingly limited, even when these ASRs share the same architecture. Existing attempts to achieve this goal have only demonstrated limited transferability [47, 68]. As demonstrated in prior work [31, 51, 60], the core reason for this limitation might be that adversarial examples tend to overfit the architecture and feature representations of the specific surrogate model, resulting in limited transferability to the target models with different architecture. Unlike image recognition models, ASR systems exhibit increased complexity and greater architectural diversity, leading to greater differences between surrogate ASRs and target ASRs.

Hence, generating highly transferable adversarial examples in the audio domain is more challenging.

4 ZQ-Attack

4.1 Problem Formulation

ZQ-Attack aims to generate transferable audio adversarial examples in the zero-query black-box scenario. Formally, given a carrier audio x and a target command t, ZQ-Attack optimizes the adversarial perturbation δ to enhance its effectiveness on various black-box target ASR systems. This optimization problem can be formulated as follows:

$$\max_{\delta} \ \mathop{\mathbb{P}}_{f \in \mathcal{F}} \left(f(x + \delta) = t \right), \tag{4}$$

where $\mathbb{P}$ represents the probability, and $\mathcal{F}$ denotes the set of all black-box target ASR systems. However, since the attacker lacks internal information about the target ASR system and cannot query it, directly solving this optimization problem is challenging.

An intuitive way to solve this problem is leveraging surrogate ASRs. However, optimizing the adversarial perturbation on a single surrogate ASR may result in overfitting to that specific model. Therefore, ZQ-Attack optimizes adversarial perturbations on multiple surrogate ASRs. We denote the set of surrogate ASRs as $\mathbb{F}$. Then, the optimization problem becomes as follows:

$$\min_{\delta} \mathcal{L}_{all}(x, \delta, t, \mathbb{F}) \quad \text{s.t. } Dis(x, x') < \epsilon, \tag{5}$$

where $\mathcal{L}_{all}$ denotes the loss on all surrogate ASRs, and the imposed constraint ensures that the optimized adversarial perturbation attains a specified level of imperceptibility.

4.2 Attack Overview

The core idea of ZQ-Attack is to collaboratively optimize the adversarial perturbation on diverse types of surrogate ASRs. Specifically, ZQ-Attack consists of three stages: surrogate ASRs selection, perturbation initialization, and sequential ensemble optimization. The workflow is illustrated in Figure 2.

Surrogate ASRs Selection. We investigate modern ASR systems and categorize them into two main types: CNN-based and Transformer-based. While CNNs are more adept at capturing local features, Transformers excel at capturing global contexts. Therefore, an intuitive approach is to select surrogate ASRs that include both CNN-based and Transformer-based architectures, ensuring that the adversarial perturbations optimized on these surrogate ASRs concurrently possess both local and global features of the target command.
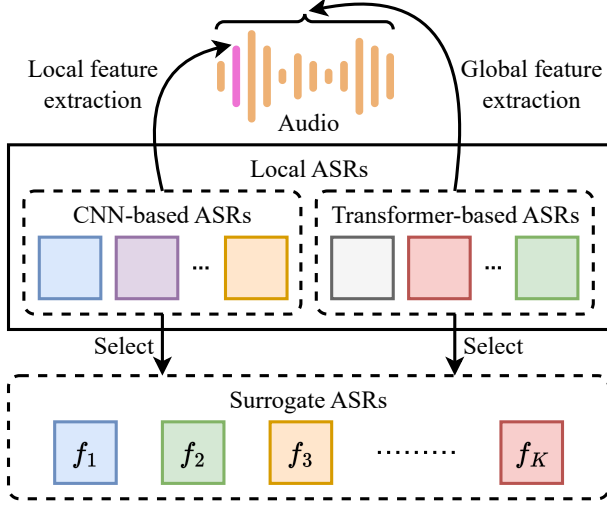
Figure 3: Illustration of surrogate ASRs selection.

these ASR systems reside in the internal acoustic model. We summarize and categorize modern ASR systems into the following two categories from the perspective of the acoustic model:

- *CNN-based.* CNNs have found widespread application in the field of computer vision and have recently exhibited notable progress in ASR as well [1, 2, 25, 35, 36, 39, 52]. The key advantages of CNNs include their model's low complexity and high computational efficiency. Additionally, CNNS are adept at extracting local features within the spectrogram.
- *Transformer-based.* These ASR systems employ Transformers as their acoustic models. Given that audio is temporal data, using recurrent neural networks (RNNs) as acoustic models is an intuitive choice [6, 26]. While RNNs can capture short-term dependencies, attention mechanisms [58] enable non-contiguous frames to attend each other, allowing Transformers to capture long-term dependencies. This results in better speech recognition performances than RNNs [33, 37, 66]. Consequently, Transformers are progressively supplanting RNNs in modern ASRs [8, 13, 23, 29, 30, 34, 50, 67].

Perturbation Initialization. Given a target command t, we employ Text-to-Speech (TTS) techniques to generate a corresponding target command audio x_t. Subsequently, we initialize the adversarial perturbation with a scaled x_t and superimpose it onto the carrier audio, ensuring that the constructed adversarial example is effective on all surrogate ASRs. To further enhance the imperceptibility of the initial adversarial perturbation, we employ an adaptive search algorithm to minimize the scaling factor.

Sequential Ensemble Optimization. Following the perturbation initialization stage, ZQ-Attack employs a sequential ensemble optimization algorithm to collaboratively optimize the adversarial perturbation on the ordered set of surrogate ASRs. This algorithm consists of an inner loop and an outer loop. In each iteration, the ordered set of surrogate ASRs is randomly shuffled in the outer loop. Then, the sequential ensemble optimization takes place within the inner loop. This algorithm integrates collaborative information from all preceding surrogate ASRs in the ordered set for each surrogate ASR, facilitating collaborative optimization. Additionally, we design a novel loss function for the optimization process to enhance the transferability and imperceptibility of the perturbation. Following the inner loop, the algorithm updates the perturbation and validates its effectiveness on all surrogate ASRs.

4.3 Surrogate ASRs Selection

The selection of surrogate ASRs is the foundation that ensures the effectiveness of ZQ-Attack. As described in Section 3.2, the architecture of ASR systems exhibits considerable diversity. Therefore, randomly selecting surrogate ASRs may fail to cover the mainstream modern ASR systems. Hence, in this subsection, we initially provide a summary and categorization of modern ASR systems, followed by the details of surrogate ASRs selection. The illustration of this stage is shown in Figure 3.

Summary and Categorization of Modern ASR Systems. Modern ASR systems typically convert audio into the spectrogram without employing additional acoustic feature extraction algorithms [6, 23, 25, 26, 35, 36, 39, 42, 67]. Hence, the primary differences among

Selection of Surrogate ASRs. The local ASRs for selection can be obtained from online sources or trained by the attackers themselves. While utilizing a single surrogate ASR can lead to the overfitting of δ to that surrogate ASR, rendering δ ineffective on target ASR systems, using too many surrogate ASRs can also lead to high computation costs. The diversity in architectures of ASRs may result in significant disparities between the surrogate ASRs and the target ASR systems, making it challenging to generate transferable adversarial examples. Therefore, we need to select surrogate ASRs encompassing different types of modern ASR systems.

According to our categorization, modern ASR systems can be mainly categorized into CNN-based and Transformer-based ASR systems. The CNNs excel in extracting local features but exhibit a comparatively weaker capability in capturing dynamic global contexts. Conversely, Transformers are proficient in effectively capturing global information but demonstrate a diminished ability to extract local features. To integrate the advantages of both CNN-based and Transformer-based ASR systems, the selected surrogate ASRs should encompass representatives from both categories. This ensures that the optimized adversarial perturbations can possess both locally and globally salient features, thereby enhancing their transferability to the target ASR systems. Furthermore, CNNs and Transformers represent prevalent architectures of acoustic models adopted in modern ASR systems. Therefore, we incorporate both CNN-based and Transformer-based ASRs into the surrogate ASRs to effectively cover a broad range of real-world ASR systems. Upon selecting K surrogate ASRs from the local ASRs, we construct the set of these surrogate ASRs, denoted as $\mathbb{F} = [f_j]_{j=1}^{K}$, where f_j represents the j-th surrogate ASR in $\mathbb{F}$. Since the subsequent sequential ensemble optimization algorithm optimizes the adversarial perturbation on these surrogate ASRs in a sequential manner, $\mathbb{F}$ is an ordered set. It is worth noting that the surrogate ASRs are scalable. The quantity of surrogate ASRs can be adjusted flexibly depending on the computational resources of the attackers.

Algorithm 1 Adaptive Search (AdaSearch)

Input: Carrier audio x, Target command t, Target command audio x_t, Ordered set of K surrogate ASRs $\mathbb{F}$, Search stride s
Output: Initial adversarial perturbation δ

1: $l_x \leftarrow len(x)$, $l_t \leftarrow len(x_t)$
2: $\mu = +\infty$, $\delta \leftarrow 0$
3: **for** $i \leftarrow 0$ **to** $l_x - l_t$ **do**
4:     $\mu_i \leftarrow 0$
5:     $\delta_i \leftarrow [\underbrace{0, \cdots, 0}_{i}, x_t, \underbrace{0, \cdots, 0}_{l_x - l_t - i}]$
6:     **while** $\mu_i \leq \mu$ **do**
7:         **if** $\forall f \in \mathbb{F}$, $f(x + \mu_i \cdot \delta_i) = t$ **then**
8:             $\mu = \mu_i$, $\delta = \mu \cdot \delta_i$
9:             **break**
10:         **end if**
11:         $\mu_i \leftarrow \mu_i + s$
12:     **end while**
13: **end for**
14: **return** δ

4.4 Perturbation Initialization

The initialization of the adversarial perturbation δ may significantly impact the performance of the attack. Initializing δ from a point far from the region of the target command in the feature space can lead to a time-consuming and uncertain optimization process, and the high dimensionality of audio data further complicates the optimization. In contrast, using the target command audio directly as the initial δ may result in poor imperceptibility. To obtain an effective and relatively imperceptible initialized adversarial perturbation, we propose an adaptive search algorithm to initialize δ with a scaled target command audio, as presented in Alg. 1. This algorithm aims to minimize the scaling factor while maintaining the effectiveness of the initialized δ on all surrogate ASRs.

Specifically, we first choose an audio x as the carrier audio to construct the adversarial example $x' = x + \delta$. Following previous works [64, 68], we opt for songs as the carrier audio. For a given target command t, we utilize TTS techniques to generate a corresponding target command audio $x_t = \mathcal{T}(t)$, where $\mathcal{T}(\cdot)$ denotes the TTS process. To alleviate the impact of varying volume levels during the perturbation initialization stage, we normalize the values of sample points in both x and x_t to the range of $[-0.5, 0.5]$. Subsequently, the adaptive search algorithm searches for the smallest value of scaling factor μ, and δ is initialized using the scaled target command audio $\mu \cdot x_t$, ensuring that the corresponding initial adversarial example is recognized by all surrogate ASRs as the target command.

Since the length of x_t is typically shorter than that of x, it is necessary to pad both sides of the scaled x_t with zeros to initialize the perturbation. We use l_x and l_t to denote the length of x and x_t, respectively. The padding lengths on each side are indeterminate, as long as their combined total equals $l_x - l_t$. The adaptive search algorithm searches for the optimal padding lengths on each side to minimize the scaling factor as much as possible. An example of this initialization method is depicted in Figure 4. It can be seen
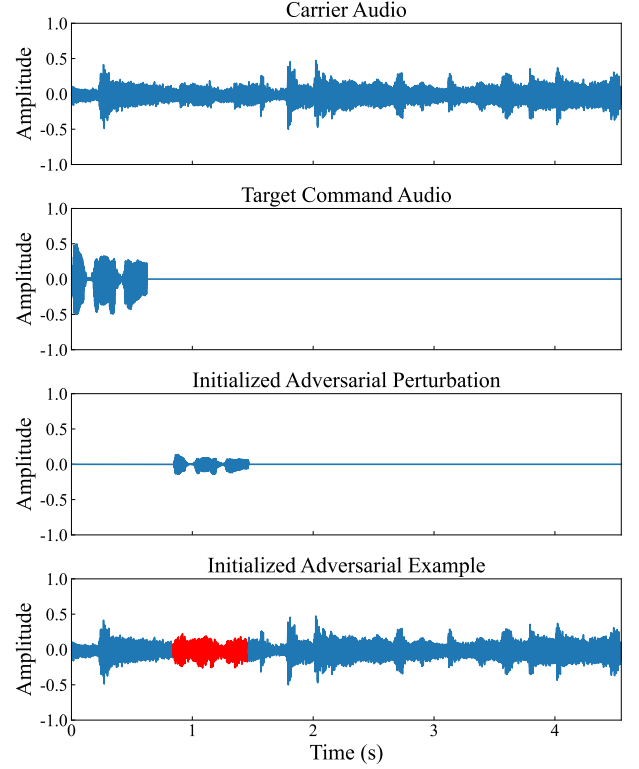

Figure 4: An example of the perturbation initialization. The adversarial perturbation is initialized using a scaled target command audio. The region of the added initialized adversarial perturbation is highlighted in red.

that the adaptive search algorithm finds the padding lengths and a relatively small scaling factor.

In summary, the adaptive search algorithm initializes δ by pushing the corresponding adversarial example toward the decision boundary of all surrogate ASRs, thereby circumventing the time-consuming and uncertain initial search process. The initialized adversarial perturbation can be regarded as a coarse-grained optimized perturbation, serving as a basis for subsequent fine-grained optimization in the sequential ensemble optimization stage.

4.5 Sequential Ensemble Optimization

After initializing the adversarial perturbation, ZQ-Attack performs fine-grained optimization of the adversarial perturbation on surrogate ASRs. Unlike the target black-box ASR systems, where the attacker lacks knowledge of their internal architectures and parameters, the white-box surrogate ASRs provide full control to the attacker. Hence, the attacker can optimize δ using any information acquired through these surrogate ASRs.

A straightforward approach to optimizing δ on diverse surrogate ASRs is to use the weighted average of the gradients from each one. For the j-th surrogate ASR $f_j \in \mathbb{F}$, the gradient is calculated as $\nabla_\delta \mathcal{L}(x, \delta, t, f_j)$, where $\mathcal{L}$ represents the loss of δ on a single surrogate ASR f_j, typically similar to Eq. (2). However, this
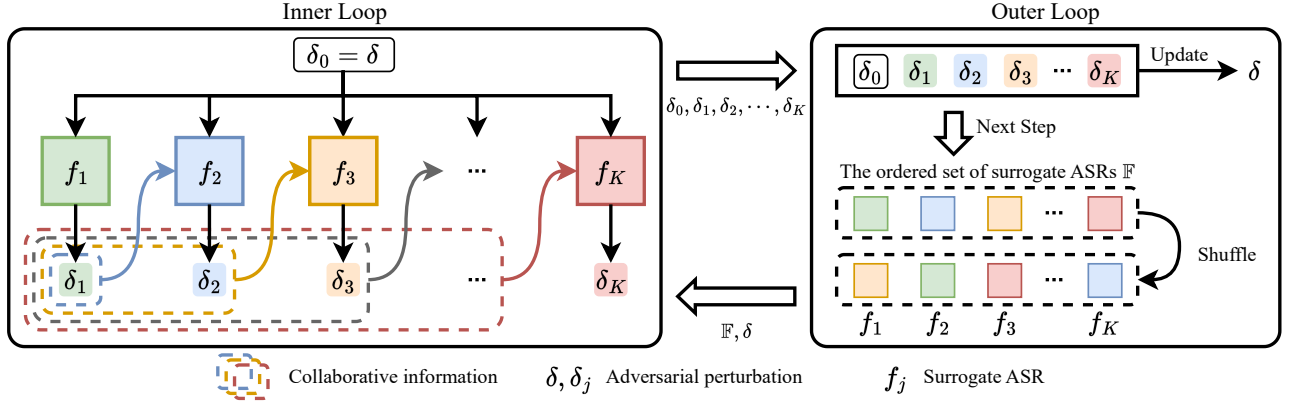
Figure 5: Illustration of sequential ensemble optimization.

method essentially treats each surrogate ASR independently when optimizing δ. Each surrogate ASR does not interact with the others and, therefore, cannot leverage the optimization information provided by others. Moreover, this method focuses on optimizing δ in the direction most effective for a single surrogate ASR, without considering the effectiveness of δ on other surrogate ASRs.

We propose a sequential ensemble optimization algorithm to facilitate collaboration among these surrogate ASRs, as presented in Figure 5. This algorithm iteratively optimizes the adversarial perturbation on the ordered set of surrogate ASRs $\mathbb{F}$. For each surrogate ASR, this algorithm leverages the collaborative information from the preceding surrogate ASRs in the ordered set to optimize δ. In other words, the optimization process considers not only the efficacy of δ on the current surrogate ASR but also its efficacy on the preceding surrogate ASRs. Additionally, instead of directly using $\mathcal{L}$ in Eq. (2), we carefully design a novel loss function comprising three loss terms. The detailed design of the loss function is presented in Section 4.6.

Specifically, the sequential ensemble optimization algorithm comprises an outer loop and an inner loop. At each step, the algorithm first randomly shuffles $\mathbb{F}$ in the outer loop. Then, the optimization of δ on $\mathbb{F}$ occurs in the inner loop. After completing the inner loop, δ is updated in the outer loop. As ZQ-Attack abstains from interacting with the target ASR systems via queries, it relies on surrogate ASRs to validate the efficacy of the generated adversarial example x'. The set of valid adversarial examples, denoted as X', is initialized as $\varnothing$ at the beginning of the algorithm. At the end of each step, x' will be added to X' if it can successfully attack all surrogate ASRs.

Inner Loop. For clarity, we use δ_j to denote the adversarial perturbation optimized on the previous j surrogate ASR(s) in $\mathbb{F}$, and δ_0 is equivalent to δ. For the j-th surrogate ASR f_j in $\mathbb{F}$, the input perturbations include δ_0 and the optimized adversarial perturbations obtained from the preceding $j-1$ ASR(s). Formally, the input adversarial perturbations for f_j, denoted as Δ_j, can be represented as $\Delta_j = [\delta_0, \delta_1, \cdots, \delta_{j-1}]$. We first add the adversarial perturbations Δ_j to the carrier audio x to construct the adversarial examples. To ensure that the adversarial examples have been pushed towards the

decision boundary, we additionally add randomly sampled Gaussian noise σ on each adversarial example. Following the forwarding of these adversarial examples to f_j and the subsequent gradient calculation, the perturbation is updated as:

$$\delta_j = \delta_0 - \alpha \cdot \frac{1}{j} \sum_{\delta' \in \Delta_j} \nabla_{\delta'} \mathcal{L}(x, \delta' + \sigma, t, f_j), \qquad (6)$$

where α represents the learning rate. The loss $\mathcal{L}$ is detailed in Section 4.6.

To avoid the updated δ_j being perceptible enough to the human ear, we use a clipping algorithm to restrict the updated δ_j within a limited range. Instead of using the L_p norm-based clipping algorithm employed in previous work [11, 68], we utilize an adaptive clipping algorithm based on psychoacoustics [19]. To be specific, this algorithm is grounded in temporal masking, a phenomenon where the presence of louder components can influence the perception of quieter components when the human ear hears two sounds with different loudness levels. This algorithm constrains δ within a range proportional to the carrier audio, permitting larger perturbations in louder segments of the carrier while maintaining smaller perturbations in quieter areas, thereby reducing the perceptibility of the perturbation. This clipping algorithm can be represented as follows:

$$clip_\epsilon(\delta, x) = \max(\min(\delta, \epsilon \cdot |x|), -\epsilon \cdot |x|). \qquad (7)$$

It is noteworthy that the clipping is performed element-wise in the perturbation. After applying the adaptive clipping algorithm, the adversarial perturbation will be bounded as $\delta_j = clip_\epsilon(\delta_j, x)$.

Outer Loop. Each step of the sequential ensemble optimization algorithm begins by randomly shuffling $\mathbb{F}$. Subsequently, the optimization is performed in the inner loop. Following the inner loop, we obtain K perturbations $[\delta_1, \delta_2, \cdots, \delta_K]$ optimized on the surrogate ASRs, and δ is updated as follows:

$$\delta = \eta \cdot \frac{1}{K} \sum_{j=1}^{K} \delta_j + (1 - \eta) \cdot \delta_0, \qquad (8)$$

where η denotes a balancing factor determining the extent of the update. $\eta = 0$ implies no update, while $\eta = 1$ signifies a complete

Algorithm 2 ZQ-Attack

Input: Carrier audio x, Target command t, Ordered set of K surrogate ASRs $\mathbb{F}$, Max step N, Search stride s

Output: The set of valid adversarial examples X'

1: $x_t \leftarrow \mathcal{T}(t)$
2: $\delta \leftarrow \text{AdaSearch}(x, t, x_t, \mathbb{F}, s)$
3: $X' \leftarrow \varnothing$
4: **# Outer Loop**
5: **for** $i \leftarrow 1$ **to** N **do**
6:     Randomly shuffle $\mathbb{F}$
7:     $\delta_0 \leftarrow \delta$
8:     **# Inner Loop**
9:     **for** $j \leftarrow 1$ **to** K **do**
10:         $f_j \leftarrow$ the j-th ASR in $\mathbb{F}$
11:         $\Delta_j \leftarrow [\delta_0, \delta_1, \delta_2, \cdots, \delta_{j-1}]$
12:         Sample a Gaussian noise σ
13:         Compute $\nabla_{\delta'} \mathcal{L}(x, \delta' + \sigma, t, f_j)$ for each δ' in Δ_j
14:         Update δ_j using Eq. (6)
15:         Clip δ_j using Eq. (7)
16:     **end for**
17:     Update δ using Eq. (8)
18:     $x' = x + \delta$
19:     **if** $\forall f \in \mathbb{F}, \ f(x') = t$ **then**
20:         $X' \leftarrow X' \cup x'$
21:     **end if**
22: **end for**
23: **return** X'

---

update to the optimized adversarial perturbation, disregarding the former one. This update method uses the adversarial perturbation δ_0 as historical information, while the momentum method [14] uses the gradient as historical information. At the end of each step, $x' = x + \delta$ is added to X' if all surrogate ASRs transcribe it into the target command.

The comprehensive details of the sequential ensemble optimization algorithm are presented in Alg. 2.

4.6 Loss Design

We design a novel loss $\mathcal{L}$ for optimizing the adversarial perturbations. It comprises three terms: the adversarial loss $\mathcal{L}_a$, the imperceptibility loss $\mathcal{L}_p$, and the acoustic feature loss $\mathcal{L}_f$. The loss $\mathcal{L}$ can be written as follows:

$$\mathcal{L}(x, \delta, t, f) = \mathcal{L}_a(x, \delta, t, f) + c_1 \cdot \mathcal{L}_p(x, \delta) + c_2 \cdot \mathcal{L}_f(x, \delta, t), \quad (9)$$

where c_1 and c_2 serve as weighting factors to balance the relative importance of different loss terms, ensuring a trade-off between the effectiveness and imperceptibility of δ.

Adversarial Loss. The adversarial loss $\mathcal{L}_a$ measures the effectiveness of δ on a surrogate ASR, evaluating how accurately a specific surrogate ASR transcribes the adversarial example to match the target command. The calculation process of $\mathcal{L}_a$ begins by inputting the constructed adversarial example into the surrogate ASR to obtain the output probability. Then, $\mathcal{L}_a$ is calculated according to the output probability and t. As different ASR systems may utilize different loss functions for training, such as connectionist temporal classification (CTC) [21] and Transducer [20, 22], the calculation

method of $\mathcal{L}_a$ varies depending on the specific surrogate ASR. For instance, we utilize CTC loss to compute $\mathcal{L}_a$ for a surrogate ASR with a CTC architecture (*e.g.*, Citrinet [39]).

Imperceptibility Loss. The imperceptibility loss $\mathcal{L}_p$ aims to minimize the detectability of the adversarial perturbation by human ears. Previous works [11, 61, 68] commonly utilize the L_p norm of the adversarial perturbation as the imperceptibility loss. However, Duan *et al.* [15] demonstrated that the L_p norm shows a limited correlation with human perception, and the L_2 norm exhibits a relatively high correlation with human perception among the L_p norm. Similar to the adaptive clipping algorithm employed when clipping the adversarial perturbation, as shown in Eq. (7), we design a new imperceptibility loss function $\mathcal{L}_p$ to calculate the L_2 norm of the ratio of the adversarial perturbation to the carrier audio. Formally, it can be represented as:

$$\mathcal{L}_p(x, \delta) = \left\| \frac{\delta}{x} \right\|_2. \quad (10)$$

Acoustic Feature Loss. As mentioned in Section 2.1, traditional ASR systems employ feature extraction algorithms to obtain the acoustic features of spectrograms as a preprocessing procedure, while modern ASR systems typically utilize the spectrograms directly. Despite the intricacy of acoustic feature extraction algorithms, which require specialized knowledge, the performance of traditional ASR systems demonstrates that these algorithms can extract high-quality features. Prior work has also demonstrated the effectiveness of incorporating acoustic features into the optimization process of audio adversarial examples [61]. Hence, we utilize the acoustic feature loss based on the widely adopted acoustic feature extraction algorithm, MFCC [54], to further enhance the effectiveness of the adversarial perturbation. Specifically, we first extract the acoustic features of the target command audio and the constructed adversarial example. We denote the acoustic feature of the target command audio and the constructed adversarial example as M_t and $M_{x'}$, respectively. Then, the acoustic feature loss $\mathcal{L}_f$ can be calculated as:

$$\mathcal{L}_f(x, \delta, t) = \| M_{x'} - M_t \|_2. \quad (11)$$

5 Experiments

5.1 Experiment Setup

In this section, we evaluate the performance of ZQ-Attack in both over-the-line and over-the-air settings. In the over-the-line setting, the audio adversarial examples are transmitted directly to the target ASR systems as waveform audio files. In the over-the-air setting, we utilize a speaker positioned near the target devices (*e.g.*, 10 cm) to play the audio adversarial examples in a quiet office environment (*e.g.*, 35dB).

Target ASR Systems. To fully demonstrate the effectiveness of ZQ-Attack, we conduct extensive experiments on various online speech recognition services, commercial IVC devices, and open-source ASRs. The details are presented as follows.

- *Online speech recognition services.* In the over-the-line setting, the target online speech recognition services include Microsoft Azure Speech Service [40], Tencent Cloud Automatic Speech

Table 2: Summary of target ASR systems in the experiments.

ASR	Type	Acoustic Model	Word Error Rate on LibriSpeech test-clean/test-other (%)
Jasper [36]	Open-source ASR	CNN	3.9/12.0
QuartzNet [35]	Open-source ASR	CNN	3.8/10.4
Citrinet [39]	Open-source ASR	CNN	4.4/10.7 (S) 3.7/8.9 (M) 3.6/7.9 (L)
ContextNet [25]	Open-source ASR	CNN	3.3/8.0 (S) 2.2/5.0 (M) 1.9/3.9 (L)
Conformer-CTC [23]	Open-source ASR	Transformer	3.7/8.1 (S) 2.6/5.9 (M) 2.1/4.5 (L) 2.0/3.7 (XL)
Conformer-Transducer [23]	Open-source ASR	Transformer	2.9/6.6 (S) 2.1/4.7 (M) 1.7/3.7 (L) 1.6/3.0 (XL)
Whisper [50]	Open-source ASR	Transformer	5.0/12.4 (base) 3.4/7.6 (small) 2.9/5.9 (medium) 2.7/5.6 (large)
Microsoft Azure [40]	Online speech recognition service	-**	-
Tencent Cloud [57]	Online speech recognition service	-	-
Alibaba Cloud [4]	Online speech recognition service	-	-
OpenAI Whisper [42] ‖	Online speech recognition service	Transformer	2.7/5.2
Apple Siri [7]	Commercial IVC device	-	-
Amazon Alexa [5]	Commercial IVC device	-	-

Note that, (i) **: Most online speech recognition services and commercial IVC devices do not reveal their implementation of the underlying ASR systems. Hence, we lack knowledge about their acoustic models and related information. We use "-" to represent unknown information. (ii) ‖: The ASR system employed by the OpenAI API is the open-source Whisper large-v2. Therefore, we have access to information regarding the acoustic model and its recognition performance on LibriSpeech. Despite the open-source nature of Whisper large-v2, we treat it as a black-box ASR system during attacks.

Recognition [57], Alibaba Cloud intelligent speech interaction [4], and OpenAI Whisper [42].

- *Commercial IVC devices.* In the over-the-air setting, we select Apple Siri [7] and Amazon Alexa [5] as the target IVC devices.
- *Open-source ASRs.* Among the numerous open-source ASRs, we select some of the most representative and advanced ASRs as our targets in the over-the-line setting. As prior research indicates that the transferability of audio adversarial examples can be limited even among instances of the same ASR [3], we select open-source ASRs with the same architectures but varying scales, as well as those with distinct architectures. Specifically, the target ASRs include: Jasper [36], QuartzNet [35], ContextNet (M/L) [25], Citrinet (M/L) [39], Conformer-CTC (M/L/XL) [23], Conformer-Transducer (M/L/XL) [23], and Whisper (base, small, medium, large) [50].

Table 2 summarizes and categorizes these target systems.

Surrogate ASRs. The surrogate ASRs include Conformer-CTC (S) [23], Conformer-Transducer (S) [23], ContextNet (S) [25], and Citrinet (S) [39]. ContextNet and Citrinet employ CNNs as the acoustic model, while the acoustic models of Conformer-CTC and Conformer-Transducer are based on Transformers. The checkpoints for these surrogate ASRs are obtained from Nvidia NeMo [41].

Target Commands and Carrier Audios. We choose 10 commonly used instructions as the target commands for the experiments: *call my wife*, *make it warmer*, *navigate to my home*, *open the door*, *open the website*, *play music*, *send a text*, *take a picture*, *turn off the light*, and *turn on airplane mode*. We select five songs used in Commandersong [64] as the carrier audio.

Software and Hardware. We implement ZQ-Attack using the PyTorch framework [46]. The experiments are conducted on a server with 8 NVIDIA 3080Ti GPUs, 2 Intel Xeon Gold 5117 CPUs, and 128 GB RAM, running a 64-bit Ubuntu 18.04 operating system. In the over-the-air setting, we use a JBL Clip3 speaker to play the audio adversarial examples. Apple Siri on an iPhone 13 and Amazon Alexa on a second-generation Amazon Echo Dot are used as the target commercial IVC devices.

Baselines. To demonstrate the superior performance of ZQ-Attack, we compare it with several previous works. In the over-the-line setting, we compare ZQ-Attack with Carlini *et al.* [10], Occam [68] and KENKU [61]. In the over-the-air setting, we compare ZQ-Attack with NI-Occam [68] and KENKU. For these baselines, we either utilize their open-source code or re-implement them. It is noteworthy that the evaluation results of these methods might be inconsistent with the original paper due to periodic updates by manufacturers to their ASR systems.

Experiment Design. We evaluate ZQ-Attack on online speech recognition services, commercial IVC devices, and open-source ASRs in Section 5.3, Section 5.4, and Section 5.5, respectively. In Section 5.6, we explore the impact of surrogate ASRs on ZQ-Attack. To evaluate the imperceptibility of ZQ-Attack, we conduct a user study in Section 5.7. In Section 5.8 and Section 5.9, we evaluate ZQ-Attack on a large command set and the latest Whisper large-v3 [43], respectively. Due to the space limitation, some detailed evaluation results and additional experiments are delegated to the full version of this paper.

Ethical Considerations. We have informed the relevant companies about the potential vulnerability of their ASR systems to our attacks via email.

5.2 Evaluation Metrics

We use the success rate of attack (SRoA) to measure the effectiveness of the attack. SRoA is calculated by dividing the number of successfully attacked commands by the total number of commands (*i.e.*, 10). For each target command, if we can generate at least one adversarial example that effectively attacks the target ASR system, we consider the attack on that command as successful. Note that the adversarial example is considered effective only when its transcription matches exactly the target command. Any word errors are regarded as a failure, with case sensitivity being disregarded.

We use the signal-to-noise ratio (SNR) to evaluate the imperceptibility of the adversarial examples. SNR is defined as the ratio of the power of a signal (*i.e.*, the carrier audio x) to the power of a

Table 3: Comparison on online speech recognition services.

Method	SRoA ↑					SNR (dB) ↑	Query ↓
	Azure	Tencent	Alibaba	OpenAI	Average		
Carlini *et al.* [10]	0/10	0/10	0/10	0/10	0/10	/	0
Occam [68]	10/10	10/10	10/10	10/10	10/10	12.54	30000
KENKU [61]	10/10	8/10	0/10	9/10	6.75/10	12.72	>0
ZQ-Attack	**10/10**	**10/10**	**10/10**	**10/10**	**10/10**	**21.91**	**0**

Table 4: Comparison on commercial IVC devices.

Method	SRoA ↑			SNR (dB) ↑
	Siri	Alexa	Average	
NI-Occam [68]	4/10	5/10	4.5/10	8.38
KENKU [61]	7/10	9/10	8/10	12.72
ZQ-Attack	**10/10**	**10/10**	**10/10**	**15.77**

noise (*i.e.*, the adversarial perturbation δ) in the logarithm scale, and a higher SNR indicates better imperceptibility. The specific calculation method is shown as follows:

$$SNR(\text{dB}) = 10 \cdot \log_{10}\left(\frac{\|x\|_2^2}{\|\delta\|_2^2}\right). \quad (12)$$

5.3 Evaluation on Online Speech Recognition Services

In the over-the-line setting, the results of ZQ-Attack and baseline methods on online speech recognition services are shown in Table 3. ZQ-Attack successfully generates audio adversarial examples for all target commands on four online speech recognition services, achieving an average SRoA of 100% and an average SNR of 21.91dB.

For baseline methods, the adversarial examples generated by Carlini *et al.* [10] fail to successfully attack any online speech recognition services, as this method is tailored for the white-box setting. Compared to Occam, ZQ-Attack achieves comparable effectiveness and better imperceptibility without any queries. While KENKU still necessitates a small number of queries to search for appropriate hyperparameters tailored to a specific target ASR system, ZQ-Attack attains superior effectiveness and imperceptibility without any queries. Note that KENKU failed to attack Alibaba successfully in our evaluation. We speculate that this could be attributed to updates made by Alibaba to its ASR system.

5.4 Evaluation on Commercial IVC Devices

In the over-the-air setting, we generate 10 audio adversarial examples for each target command. Each adversarial example is played up to three times. We consider that the attack on a command is successful if at least one adversarial example is effective.

The results of ZQ-Attack and baseline methods on commercial IVC devices are presented in Table 4. ZQ-Attack successfully generates audio adversarial examples for all target commands on two commercial IVC devices, achieving an SRoA of 100% with an average SNR of 15.77dB. Additionally, an average of 6.6 adversarial examples for each command are effective.

In comparison, the SRoA of ZQ-Attack surpasses KENKU and NI-Occam by 20% and 55%. Concurrently, adversarial examples generated by ZQ-Attack exhibit better imperceptibility, with SNR values exceeding those of KENKU and NI-Occam by 3.05dB and 7.39dB, respectively.

5.5 Evaluation on Open-source ASRs

To demonstrate the high transferability of audio adversarial examples generated by ZQ-Attack, we additionally evaluate ZQ-Attack on 16 open-source ASR systems. The evaluation results are shown in Table 5. ZQ-Attack successfully generates audio adversarial examples on all 16 open-source ASRs, achieving an average SRoA of 100% and an average SNR of 19.67dB. These results show that the audio adversarial examples generated by ZQ-Attack exhibit high transferability, successfully attacking the target open-source ASRs.

Additionally, we observe a potential positive correlation between the recognition performance of open-source ASRs and the success rate of transfer attacks. We speculate that this phenomenon arises due to the better feature extraction capabilities of a more powerful ASR system, facilitating the capture of subtle adversarial perturbations. The detailed evaluation results and analysis of the correlation can be found in the full version of this paper [16].
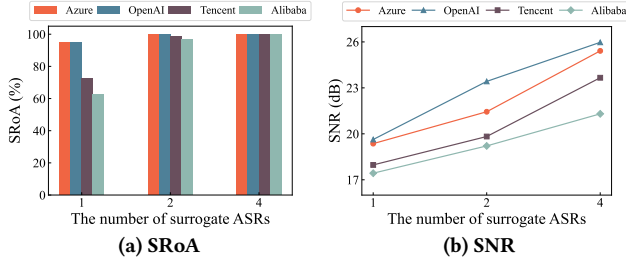
5.6 Impact of Surrogate ASRs

ZQ-Attack optimizes the adversarial perturbation on an ordered set of K surrogate ASRs. To investigate the impact of surrogate ASRs on the performance of ZQ-Attack, we conduct experiments with various values of K. Specifically, we set K to 1, 2, and 4, respectively. In the cases with only one surrogate ASR, we conduct experiments with four configurations: ContextNet, Citrinet, Conformer-CTC, and Conformer-Transducer (*i.e.*, selecting one from the four available ASRs). In the cases with two surrogate ASRs, six configurations (*i.e.*, combination of two from the four available ASRs) are explored. Subsequently, we compute the average SRoA and SNR for the three groups of experiments. In these experiments, we choose one carrier audio and generate audio adversarial examples for 10 target commands.

The results are presented in Figure 6. It can be observed that with an increase in K, both SRoA and SNR exhibit an increasing trend. For instance, when K is 1, 2, and 4, the average SRoA is 81.25%, 98.75%, and 100%, respectively. The average SNR for the case involving 4 surrogate ASRs surpasses that of the case with 2 surrogate ASRs by 2.98dB and exceeds the SNR of the case with only 1 surrogate ASR by 5.35dB. Additionally, it can also be observed that ZQ-Attack successfully performs attacks for most target commands when using 2 surrogate ASRs. Thus, when computational resources

Table 5: Evaluation on open-source ASRs.

Target ASR	SRoA	SNR (dB)	Target ASR	SRoA	SNR (dB)
Jasper	10/10	13.59	Conformer-CTC (XL)	10/10	23.59
QuartzNet	10/10	12.96	Conformer-Transducer (M)	10/10	25.34
Citrinet (M)	10/10	14.67	Conformer-Transducer (L)	10/10	20.63
Citrinet (L)	10/10	15.89	Conformer-Transducer (XL)	10/10	21.08
ContextNet (M)	10/10	15.73	Whisper (base)	10/10	17.88
ContextNet (L)	10/10	16.87	Whisper (small)	10/10	20.76
Conformer-CTC (M)	10/10	25.13	Whisper (medium)	10/10	23.39
Conformer-CTC (L)	10/10	23.51	Whisper (large)	10/10	23.74

Figure 6: Impact of Surrogate ASRs.

are constrained, and there is a more relaxed requirement for the imperceptibility and effectiveness of the attack, a smaller K can be chosen. Conversely, a larger K can be used. In the cases with four surrogate ASRs, ZQ-Attack can complete a generation process in approximately 10 minutes using only one NVIDIA 3080Ti GPU.

5.7 User Study

In this section, we recruit human participants to analyze the imperceptibility of the audio adversarial examples and conduct a comparative analysis between ZQ-Attack and the baseline methods (*i.e.*, Occam, and KENKU). This study is carefully designed to mitigate any conceivable risks (psychological, legal, etc.) to the participants, and it is approved by the Institutional Review Board (IRB). The target commands are common household phrases (*e.g.*, "call my wife") to minimize discomfort, and the audio volume is normalized to maintain it below a safe threshold, preventing any risk of hearing damage. Besides, our study does not collect any private information from the participants, and all data will be deleted upon the completion of the study.

Our test audios comprise both normal audios and audio adversarial examples. The normal audios consist of 10 songs, and the audio adversarial examples include 10 instances from each method. In the user study, each participant listens to all test audios. Following the first listening, participants are provided with three choices: "normal audio", "noisy audio", and "audio with background speech". In cases where participants select the "audio with background speech" option, they are required to provide the content of the perceived speech, followed by a second round of listening and transcription of the same test audio. Specifically, participants first listen to the test audios through speakers (*e.g.*, MacBook Pro Speaker). Then, to eliminate environmental interference, participants listen to the

audios again through noise-canceling headphones (*e.g.*, Sony WH-1000XM5).

We gather data from a total of 38 participants, consisting of 20 males and 18 females. Among this group, 10 participants are below the age of 22, 18 participants are aged 22 to 24, and 10 participants are 25 or older. All participants are proficient in both spoken and written English, hold at least a bachelor's degree, and have normal hearing. The results are presented in Table 6. When using the speaker as the audio device, 13.16% of participants select our adversarial examples as "normal audio", while 71.58% of participants select them as "noisy audio". Although 15.26% of participants select our adversarial examples as "audio with background speech," only 3.68% of commands are recognized even after a second round of listening and transcription. In a more challenging configuration that uses headphones to play the audios, only 22.63% of the commands are recognized even after a second round of listening and transcription. Additionally, for the audios played using speakers and headphones, the average word error rate (WER) between user recognition results and the actual target commands is 94.47% and 72.41%, respectively. Compared with other methods, ZQ-Attack attains superior imperceptibility.

Furthermore, we observed that the imperceptibility of different target commands varies. The command *open the door* is the most easily perceived, while *send a text* is the hardest to perceive. This disparity may be attributed to the varying phonemes in different command audios, with vowel phonemes containing more energy and thus being more easily audible [24, 63].

5.8 Evaluation on a Large Command Set

To thoroughly evaluate the effectiveness of ZQ-Attack, we conduct experiments using a larger command set. This set includes a total of 20 commands: *ask me a question, clear notification, close the shades find a hotel, good morning, I have a secret to tell you, I need help, open the box, read a book, record a video, reset password, show me my message, show me the money, start recording, tell me a story, turn off the fan, turn on the TV, watch TV, what time is it where is my car.*

We choose a carrier audio and generate audio adversarial examples for each of these target commands. As illustrated in Table 7, ZQ-Attack can generate effective audio adversarial examples for each target command in this set, achieving an average SRoA of 100%. On Azure, Tencent, Alibaba, and OpenAI, ZQ-Attack attains average SNR values of 24.75dB, 23.99dB, 22.35dB, and 26.45dB, respectively.

Table 6: User study on human perception.

Audio Device	Method	Normal (%) ↑	Noise (%)	Talking (%) ↓	Recognize (%) ↓		WER (%) ↑	
					Once	Twice	Once	Twice
Speaker	Songs	92.63	7.37	0.00	0.00	0.00	0.00	0.00
	Occam [68]	5.26	15.79	78.95	29.47	35.79	57.37	51.24
	KENKU [61]	0.00	6.32	93.68	56.84	63.68	30.23	25.96
	ZQ-Attack	13.16	71.58	15.26	3.16	3.68	94.95	94.47
Headphone	Songs	92.11	7.89	0.00	0.00	0.00	0.00	0.00
	Occam [68]	2.11	3.16	94.74	49.47	54.74	38.25	31.32
	KENKU [61]	0.00	1.58	98.42	67.37	70.00	19.34	17.77
	ZQ-Attack	5.79	47.89	46.32	15.26	22.63	77.98	72.41

Table 7: Evaluation on a larger command set.

Command	Azure		Tencent		Alibaba		OpenAI	
	Attack	SNR (dB)	Attack	SNR (dB)	Attack	SNR (dB)	Attack	SNR (dB)
ask me a question	✓	23.51	✓	28.66	✓	26.31	✓	28.73
clear notification	✓	26.52	✓	21.28	✓	20.05	✓	26.52
close the shades	✓	26.54	✓	26.24	✓	26.54	✓	26.86
find a hotel	✓	25.78	✓	24.63	✓	20.26	✓	28.21
good morning	✓	19.65	✓	18.96	✓	27.90	✓	25.76
I have a secret to tell you	✓	27.21	✓	27.21	✓	27.21	✓	27.21
I need help	✓	27.23	✓	26.64	✓	20.17	✓	28.54
open the box	✓	18.48	✓	26.34	✓	18.48	✓	27.54
read a book	✓	20.77	✓	25.81	✓	17.27	✓	27.71
record a video	✓	21.38	✓	20.51	✓	21.38	✓	21.38
reset password	✓	22.70	✓	21.69	✓	19.91	✓	22.70
show me my message	✓	27.12	✓	27.54	✓	23.98	✓	27.54
show me the money	✓	27.94	✓	27.96	✓	25.75	✓	27.96
start recording	✓	27.03	✓	20.42	✓	19.87	✓	27.19
tell me a story	✓	27.89	✓	27.89	✓	24.34	✓	27.94
turn off the fan	✓	19.98	✓	15.25	✓	19.98	✓	19.98
turn on the TV	✓	25.55	✓	17.71	✓	17.18	✓	25.55
watch TV	✓	26.32	✓	26.32	✓	19.20	✓	26.32
what time is it	✓	26.22	✓	25.39	✓	25.39	✓	28.20
where is my car	✓	27.24	✓	23.28	✓	25.76	✓	27.24
Average	20/20	24.75	20/20	23.99	20/20	22.35	20/20	26.45

5.9 Evaluation on Whisper large-v3

Whisper [42] is a popular ASR system trained on a diverse audio dataset comprising 680,000 hours of audio, achieving state-of-the-art multilingual recognition performance. Evaluating ZQ-Attack on Whisper can more comprehensively demonstrate its effectiveness on state-of-the-art ASR systems. Since the models of Whisper come in diverse sizes, we also conduct experiments on these various models. In Section 5.5, we evaluate ZQ-Attack utilizing the base, small, medium, and large models as open-source ASRs. As the OpenAI API employs the large-v2 model, we have evaluated the performance of ZQ-Attack on the large-v2 model in Section 5.3.

Recently, OpenAI introduced the latest large-v3 model [43], which surpasses the performance of the large-v2 model. We have also evaluated ZQ-Attack on the latest model. As indicated in Table 8, ZQ-Attack can successfully generate effective adversarial examples for all 10 target commands, achieving an average SNR of 23.49dB. These results demonstrate the capability of ZQ-Attack to generate effective and imperceptible audio adversarial examples on the most advanced ASR systems.

6 Related Work

Audio Adversarial Attacks on White-box ASR Systems. In recent years, extensive studies have focused on audio adversarial attacks on ASR systems. Carlini *et al.* [10] were the first to generate targeted audio adversarial examples on the white-box DeepSpeech model. Concurrently, Commandersong [64] demonstrated successful attacks on the Kaldi model by embedding malicious commands into songs. This approach also enabled physical attacks but had stringent limitations, such as requiring specific speakers and recording devices. Qin *et al.* [48] and Schönherr *et al.* [53] contributed to

Table 8: Evaluation on the latest Whisper large-v3.

Command	Whisper large-v3	
	Attack	SNR (dB)
call my wife	✓	21.28
make it warmer	✓	17.31
navigate to my home	✓	24.35
open the door	✓	24.46
open the website	✓	26.75
play music	✓	24.76
send a text	✓	24.54
take a picture	✓	27.23
turn off the light	✓	26.29
turn on airplane mode	✓	17.95
Average	10/10	23.49

improving the imperceptibility of generated adversarial examples by incorporating the psychoacoustic model into the audio adversarial example generation process.

Audio Adversarial Attacks on Black-box ASR Systems. Despite the success of the aforementioned methods in generating audio adversarial examples on white-box ASR systems, their reliance on the gradient information of the target model limits their applicability for attacking black-box ASR systems. To address this challenge, Taori *et al.* [56] proposed employing gradient estimation and genetic algorithms to achieve black-box attacks, but their method had a relatively low attack success rate. Following this work, SGEA [59] improved the attack success rate and reduced the number of queries by employing selective gradient estimation techniques. Nevertheless, generating a single audio adversarial example still required approximately 100,000 queries. Devil's Whisper [11] significantly improved the attack success rate by using a surrogate model. However, it relied on confidence scores returned by the target ASR system, often unavailable in real-world scenarios. Occam [68] employed cooperative co-evolution and the CMA evolution strategy [27], eliminating the need for confidence scores. Recently, KENKU [61] optimized the acoustic feature loss based on MFCC and imperceptibility loss simultaneously to generate relatively stealthy audio adversarial examples on black-box ASR systems. Although these methods have achieved improved performance, their dependence on querying the target system continues to limit their practicality.

To generate audio adversarial examples without the need for queries, researchers have proposed transfer-based attacks that can generate adversarial examples capable of attacking different models. NI-Occam [68] utilized fine-tuned Kaldi models to exclusively launch attacks on IVC devices that are more sensitive to commands, but its attack success rate remains relatively low. TransAudio [47] presented a two-stage framework and a score-matching-based optimization strategy to achieve word-level adversarial attack, but its target transcription is constrained by the carrier audio. In the field of image adversarial attacks, previous work [9, 38] proposed utilizing the ensemble method to improve the transferability of adversarial examples. Inspired by this, we design a sequential ensemble optimization algorithm to generate adversarial examples using diverse surrogate ASRs.

7 Discussion

7.1 Defenses against ZQ-Attack

We evaluate ZQ-Attack with several state-of-the-art defense methods against audio adversarial attacks, including local smoothing, downsampling, temporal dependency [62], and MVP-EARS [65]. The results are presented in Table 9.

Local Smoothing. Local smoothing renders audio adversarial attacks ineffective by applying a sliding window with a median filter to the adversarial examples. Given the length of the sliding window, denoted as h, the value of an audio sample point is replaced by the average of its own value and the values of the h sample points before and after it. To evaluate the robustness of ZQ-Attack against local smoothing, we set h to 1, 2, and 3, respectively. The results in Table 9 show that local smoothing has minimal impact on ZQ-Attack. Across different settings, ZQ-Attack consistently achieves an SRoA higher than 97.5%.

Downsampling. This method involves downsampling the original audio to a lower sampling rate f_{low} and subsequently upsampling it to the original sampling rate, which causes a loss of high-frequency information from the original audio. Therefore, the attack might fail if the high-frequency information in the adversarial perturbation is lost. To assess the robustness of ZQ-Attack against downsampling, we set f_{low} to 14kHz, 12kHz, and 10kHz. The results in Table 9 show that ZQ-Attack has great robustness to downsampling, achieving an SRoA higher than 95% under different settings.

Temporal Dependency. The inherent temporal dependency in audio data can be leveraged to detect audio adversarial examples [62]. Specifically, audio adversarial examples can be identified by comparing the transcription of the first k part of the audio with the first k part of the transcription of the entire audio, where k is a ratio between 0 and 1. If their consistency is low, the audio can be considered an adversarial example; otherwise, it is accepted as normal. To evaluate the robustness of ZQ-Attack against the temporal dependency-based defense, we set k to 0.2, 0.5, and 0.8. The results in Table 9 demonstrate that the audio adversarial examples generated by ZQ-Attack exhibit strong resilience to temporal dependency, as all attacks are successful under different settings.

MVP-EARS [65]. MVP-EARS utilizes multiple ASR systems to detect audio adversarial examples. Due to the limited transferability of most prior audio adversarial example generation methods, different ASR systems may produce significantly different transcriptions for the same audio adversarial example. Therefore, multiple ASR systems can be used to transcribe the same audio. If their transcripts differ, the audio can be considered an adversarial example.

To evaluate the robustness of ZQ-Attack against MVP-EARS, we set m to 2, 3, and 4, where m is the number of ASR systems used. The results in Table 9 show that the audio adversarial examples generated by ZQ-Attack exhibit good robustness to MVP-EARS, with all target commands successfully attacked under different settings. This is attributed to the fact that ZQ-Attack does not generate audio adversarial examples customized for a specific ASR

Table 9: SRoA of ZQ-Attack on online speech recognition services with defenses.

Defenses	Setting	Azure	Tencent	Alibaba	OpenAI	Average
Local Smoothing	$h = 1$	10/10	10/10	10/10	10/10	10/10
	$h = 2$	10/10	10/10	10/10	10/10	10/10
	$h = 3$	10/10	10/10	9/10	10/10	9.75/10
Downsampling	$f_{low} = 14\text{kHz}$	10/10	10/10	10/10	10/10	10/10
	$f_{low} = 12\text{kHz}$	10/10	10/10	9/10	9/10	9.5/10
	$f_{low} = 10\text{kHz}$	10/10	10/10	9/10	9/10	9.5/10
Temporal Dependency	$k = 0.2$	10/10	10/10	10/10	10/10	10/10
	$k = 0.5$	10/10	10/10	10/10	10/10	10/10
	$k = 0.8$	10/10	10/10	10/10	10/10	10/10
MVP-EARS	$m = 2$	10/10	10/10	10/10	10/10	10/10
	$m = 3$	10/10	10/10	10/10	10/10	10/10
	$m = 4$	10/10	10/10	10/10	10/10	10/10

but crafts transferable adversarial examples on diverse surrogate ASRs.

7.2 Limitations and Future Work

While ZQ-Attack leverages diverse surrogate ASRs to achieve transferable audio adversarial attacks in the zero-query black-box setting, it incurs a higher computation cost (*e.g.*, GPU memory). Even so, as the audio adversarial examples generated by ZQ-Attack are effective on multiple ASR systems and save the cost of queries, we consider the additional cost acceptable. Another limitation is that although the audio adversarial examples generated by ZQ-Attack exhibit better imperceptibility compared to prior work in the over-the-air setting, they may still be detected by humans. We leave enhancing the imperceptibility of audio adversarial examples in the over-the-air setting for future work.

8 Conclusion

We proposed ZQ-Attack, a transfer-based adversarial attack on ASR systems in the zero-query black-box scenario. By summarizing and categorizing the modern ASR systems, we first selected a diverse set of surrogate ASRs to generate adversarial examples. Then, we employed an adaptive search algorithm to initialize the adversarial perturbations with a scaled target command audio, ensuring its effectiveness and imperceptibility. Subsequently, we designed a novel sequential ensemble optimization algorithm to optimize the adversarial perturbations using the selected surrogate ASRs. Our experimental results indicate that ZQ-attack successfully attacks 4 online speech recognition services and 16 open-source ASRs in the over-the-line setting and attacks 2 commercial IVC devices in the over-the-air setting. This demonstrates a significant improvement in the practicality of audio adversarial attacks compared to prior methods.

Acknowledgments

References

[1] Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu. 2014. Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on audio, speech, and language processing* 22 (2014), 1533–1545.

[2] Osama Abdeljaber, Onur Avci, Serkan Kiranyaz, Moncef Gabbouj, and Daniel J Inman. 2017. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of Sound and Vibration* 388 (2017), 154–170.

[3] Hadi Abdullah, Kevin Warren, Vincent Bindschaedler, Nicolas Papernot, and Patrick Traynor. 2021. Sok: The faults in our asrs: An overview of attacks against automatic speech recognition and speaker identification systems. In *Proc. of IEEE S&P*. 730–747.

[4] Alibaba. 2023. Alibaba Cloud Intelligent Speech Interaction. https://www.alibabacloud.com/product/intelligent-speech-interaction.

[5] Amazon. 2023. Alexa. https://www.alexa.com/.

[6] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *Proc. of ICML*. 173–182.

[7] Apple. 2023. Siri. https://www.apple.com/siri/.

[8] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems* 33 (2020), 12449–12460.

[9] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. 2017. Adversarial patch. *CoRR* abs/1712.09665 (2017).

[10] Nicholas Carlini and David Wagner. 2018. Audio adversarial examples: Targeted attacks on speech-to-text. In *Proc. of IEEE SPW*. 1–7.

[11] Yuxuan Chen, Xuejing Yuan, Jiangshan Zhang, Yue Zhao, Shengzhi Zhang, Kai Chen, and XiaoFeng Wang. 2020. Devil's Whisper: A general approach for physical adversarial attacks against commercial black-box speech recognition devices.. In *Proc. of USENIX Security*. 2667–2684.

[12] Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli. 2019. Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In *Proc. of USENIX Security*. 321–338.

[13] Linhao Dong, Shuang Xu, and Bo Xu. 2018. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *Proc. of IEEE ICASSP*. 5884–5888.

[14] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting adversarial attacks with momentum. In *Proc. of IEEE/CVF CVPR*. 9185–9193.

[15] Rui Duan, Zhe Qu, Shangqing Zhao, Leah Ding, Yao Liu, and Zhuo Lu. 2022. Perception-aware attack: Creating adversarial music via reverse-engineering human perception. In *Proc. of ACM CCS*. 905–919.

[16] Zheng Fang, Tao Wang, Lingchen Zhao, Shenyi Zhang, Bowen Li, Yunjie Ge, Qi Li, Chao Shen, and Qian Wang. 2024. Zero-query adversarial attack on black-box automatic speech recognition systems. *CoRR* abs/2406.19311 (2024).

[17] Mark Gales, Steve Young, et al. 2008. The application of hidden Markov models in speech recognition. *Foundations and Trends® in Signal Processing* 1 (2008), 195–304.

[18] Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *Proc. of IEEE SPW*. 50–56.

[19] S.A. Gelfand. 2017. *Hearing: An introduction to psychological and physiological acoustics*. CRC Press.

[20] Alex Graves. 2012. Sequence transduction with recurrent neural networks. *CoRR* abs/1211.3711 (2012).

[21] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. of ICML*. 369–376.

[22] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Proc. of IEEE ICASSP*. 6645–6649.

[23] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. 2020. Conformer: Convolution-augmented transformer for speech recognition. In *Proc. of INTERSPEECH*. 5036–5040.

[24] Hanqing Guo, Yuanda Wang, Nikolay Ivanov, Li Xiao, and Qiben Yan. 2022. SPECPATCH: Human-In-The-Loop Adversarial Audio Spectrogram Patch Attack on Speech Recognition. In *Proc. of ACM CCS*. 1353–1366.

[25] Wei Han, Zhengdong Zhang, Yu Zhang, Jiahui Yu, Chung-Cheng Chiu, James Qin, Anmol Gulati, Ruoming Pang, and Yonghui Wu. 2020. ContextNet: Improving convolutional neural networks for automatic speech recognition with global context. In *Proc. of INTERSPEECH*. 3610–3614.

[26] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *CoRR* abs/1412.5567 (2014).

[27] Nikolaus Hansen. 2016. The CMA evolution strategy: A tutorial. *CoRR* abs/1604.00772 (2016).

[28] Hynek Hermansky. 1990. Perceptual linear predictive (PLP) analysis of speech. *the Journal of the Acoustical Society of America* 87 (1990), 1738–1752.

[29] Yosuke Higuchi, Shinji Watanabe, Nanxin Chen, Tetsuji Ogawa, and Tetsunori Kobayashi. 2020. Mask CTC: Non-autoregressive end-to-end ASR with CTC and mask predict. In *Proc. of INTERSPEECH*. 3655–3659.

[30] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 3451–3460.

[31] Qian Huang, Isay Katsman, Horace He, Zeqi Gu, Serge Belongie, and Ser-Nam Lim. 2019. Enhancing adversarial example transferability with an intermediate level attack. In *Proc. of IEEE/CVF ICCV*. 4733–4742.

[32] Fumitada Itakura. 1975. Line spectrum representation of linear predictor coefficients of speech signals. *The Journal of the Acoustical Society of America* 57 (1975), S35–S35.

[33] Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyan Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, et al. 2019. A comparative study on transformer vs rnn in speech applications. In *Proc. of IEEE ASRU*. 449–456.

[34] Shigeki Karita, Nelson Enrique Yalta Soplin, Shinji Watanabe, Marc Delcroix, Atsunori Ogawa, and Tomohiro Nakatani. 2019. Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration. In *Proc. of INTERSPEECH*. 1408–1412.

[35] Samuel Kriman, Stanislav Beliaev, Boris Ginsburg, Jocelyn Huang, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, and Yang Zhang. 2020. Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions. In *Proc. of IEEE ICASSP*. 6124–6128.

[36] Jason Li, Vitaly Lavrukhin, Boris Ginsburg, Ryan Leary, Oleksii Kuchaiev, Jonathan M. Cohen, Huyen Nguyen, and Ravi Teja Gadde. 2019. Jasper: An end-to-end convolutional neural acoustic model. In *Proc. of INTERSPEECH*. 71–75.

[37] Jinyu Li, Yu Wu, Yashesh Gaur, Chengyi Wang, Rui Zhao, and Shujie Liu. 2020. On the comparison of popular end-to-end models for large scale speech recognition. In *Proc. of INTERSPEECH*. 1–5.

[38] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. 2017. Delving into transferable adversarial examples and black-box attacks. In *Proc. of ICLR*.

[39] Somshubra Majumdar, Jagadeesh Balam, Oleksii Hrinchuk, Vitaly Lavrukhin, Vahid Noroozi, and Boris Ginsburg. 2021. Citrinet: Closing the gap between non-autoregressive and autoregressive end-to-end models for automatic speech recognition. *CoRR* abs/2104.01721 (2021).

[40] Microsoft. 2023. Microsoft Azure Speech Service. https://azure.microsoft.com/en-us/products/cognitive-services/speech-services/.

[41] Nvidia. 2023. NeMo. https://developer.nvidia.com/nemo/.

[42] OpenAI. 2023. Whisper. https://openai.com/research/whisper.

[43] OpenAI. 2023. Whisper large-v3. https://github.com/openai/whisper/.

[44] David Opitz and Richard Maclin. 1999. Popular ensemble methods: An empirical study. *Journal of artificial intelligence research* 11 (1999), 169–198.

[45] Nicolas Papernot, Patrick D. McDaniel, and Ian J. Goodfellow. 2016. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *CoRR* abs/1605.07277 (2016).

[46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Proc. of NeruIPS*. 8024–8035.

[47] Gege Qi, Yuefeng Chen, Yao Zhu, Binyuan Hui, Xiaodan Li, Xiaofeng Mao, Rong Zhang, and Hui Xue. 2023. Transaudio: Towards the transferable adversarial audio attack via learning contextualized perturbations. In *Proc. of IEEE ICASSP*. 1–5.

[48] Yao Qin, Nicholas Carlini, Garrison Cottrell, Ian Goodfellow, and Colin Raffel. 2019. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *Proc. of ICML*. 5231–5240.

[49] Lawrence R Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE* 77 (1989), 257–286.

[50] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proc. of ICML*. 28492–28518.

[51] Deepak Ravikumar, Sangamesh D Kodge, Isha Garg, and Kaushik Roy. 2022. TREND: Transferability based robust ensemble design. *IEEE Transactions on Artificial Intelligence* (2022).

[52] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. In *Proc. of INTERSPEECH*. 3465–3469.

[53] Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. 2019. Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. In *Proc. of NDSS*.

[54] Sigurdur Sigurdsson, Kaare Brandt Petersen, and Tue Lehn-Schiøler. 2006. Mel Frequency Cepstral Coefficients: An Evaluation of Robustness of MP3 Encoded Music.. In *ISMIR*. 286–289.

[55] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *Proc. of ICLR*.

[56] Rohan Taori, Amog Kamsetty, Brenton Chu, and Nikita Vemuri. 2019. Targeted adversarial examples for black box audio systems. In *Proc. of IEEE SPW*. 15–20.

[57] Tencent. 2023. Tencent Cloud Automatic Speech Recognition. https://cloud.tencent.com/document/product/1093/.

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Proc. of NeruIPS*. 5998–6008.

[59] Qian Wang, Baolin Zheng, Qi Li, Chao Shen, and Zhongjie Ba. 2020. Towards query-efficient adversarial attacks against automatic speech recognition systems. *IEEE Transactions on Information Forensics and Security* 16 (2020), 896–908.

[60] Lei Wu, Zhanxing Zhu, Cheng Tai, and Weinan E. 2018. Understanding and enhancing the transferability of adversarial examples. *CoRR* abs/1802.09707 (2018).

[61] Xinghui Wu, Shiqing Ma, Chao Shen, Chenhao Lin, Qian Wang, Qi Li, and Yuan Rao. 2023. KENKU: Towards efficient and stealthy black-box adversarial attacks against ASR systems. In *Proc. of USENIX Security*. 247–264.

[62] Zhuolin Yang, Bo Li, Pin-Yu Chen, and Dawn Song. 2019. Characterizing audio adversarial examples using temporal dependency. In *Proc. of ICLR*.

[63] Zhiyuan Yu, Yuanhaur Chang, Ning Zhang, and Chaowei Xiao. 2023. {SMACK}: Semantically meaningful adversarial audio attack. In *Proc. of USENIX security*. 3799–3816.

[64] Xuejing Yuan, Yuxuan Chen, Yue Zhao, Yunhui Long, Xiaokang Liu, Kai Chen, Shengzhi Zhang, Heqing Huang, XiaoFeng Wang, and Carl A. Gunter. 2018. CommanderSong: A systematic approach for practical adversarial voice recognition. In *Proc. of USENIX Security*. 49–64.

[65] Qiang Zeng, Jianhai Su, Chenglong Fu, Golam Kayas, Lannan Luo, Xiaojiang Du, Chiu C Tan, and Jie Wu. 2019. A multiversion programming inspired approach to detecting audio adversarial examples. In *Proc. of IEEE/IFIP DSN*. 39–51.

[66] Albert Zeyer, Parnia Bahar, Kazuki Irie, Ralf Schlüter, and Hermann Ney. 2019. A comparison of transformer and lstm encoder decoder models for asr. In *Proc. of IEEE ASRU*. 8–15.

[67] Qian Zhang, Han Lu, Hasim Sak, Anshuman Tripathi, Erik McDermott, Stephen Koo, and Shankar Kumar. 2020. Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss. In *Proc. of IEEE ICASSP*. 7829–7833.

[68] Baolin Zheng, Peipei Jiang, Qian Wang, Qi Li, Chao Shen, Cong Wang, Yunjie Ge, Qingyang Teng, and Shenyi Zhang. 2021. Black-box adversarial attacks on commercial speech platforms with minimal information. In *Proc. of ACM CCS*. 86–107.