SIEVE: Secure In-Vehicle Automatic Speech Recognition Systems

Shu Wang[1], Jiahao Cao[1,2], Kun Sun[1], and Qi Li[2,3]

[1]Center for Secure Information Systems, George Mason University, Fairfax, VA, USA
[2]Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing, China
[3]Beijing National Research Center for Information Science and Technology, Beijing, China

Abstract

Driverless vehicles are becoming an irreversible trend in our daily lives, and humans can interact with cars through in-vehicle voice control systems. However, the automatic speech recognition (ASR) module in the voice control systems is vulnerable to adversarial voice commands, which may cause unexpected behaviors or even accidents in driverless cars. Due to the high demand on security insurance, it remains as a challenge to defend in-vehicle ASR systems against adversarial voice commands from various sources in a noisy driving environment. In this paper, we develop a secure in-vehicle ASR system called SIEVE, which can effectively distinguish voice commands issued from the driver, passengers, or electronic speakers in three steps. First, it filters out multiple-source voice commands from multiple vehicle speakers by leveraging an autocorrelation analysis. Second, it identifies if a single-source voice command is from humans or electronic speakers using a novel dual-domain detection method. Finally, it leverages the directions of voice sources to distinguish the voice of the driver from those of the passengers. We implement a prototype of SIEVE and perform a real-world study under different driving conditions. Experimental results show SIEVE can defeat various adversarial voice commands over in-vehicle ASR systems.

1 Introduction

Driverless cars, also known as autonomous cars or self-driving cars, are no longer the things we would only see in sci-fi films, but they are becoming an irreversible trend in our daily lives, particularly due to the rapid development of sensor techniques and advanced artificial intelligence algorithms. For instance, MIT developed a human-centered autonomous vehicle using multiple sensors and deep neural networks in 2018 [1]. From April 2019, all new Tesla cars come standard with Autopilot, providing the capability of switching modes between manual driving and self-driving [2]. Waymo's driverless cars have driven 20 million miles on public roads by January 2020 [3].

The latest in-vehicle voice control system [4] provides a convenient way for drivers and passengers to interact with driverless cars. For example, we can use our voice to control in-vehicle entertainment systems, set destinations to the GPS navigation system, and take back the full control of cars from the self-driving mode [5]. However, the core module of in-vehicle voice control system, i.e., automatic speech recognition (ASR) module, is vulnerable to various adversarial voice command attacks [6–14]. Particularly, since most in-vehicle ASR systems support speaker-independent recognition by default [15], passengers can voice malicious commands to ASR systems and thus control critical in-vehicle systems. Moreover, remote attackers may hide a voice command into a song [9]. When the song is played through car loudspeakers or smartphones' speakers, the malicious voice command in the song can be recognized by ASR systems. It may cause unexpected behaviors or even accidents in driverless cars.

A number of countermeasures [16–18] have been proposed to defend ASR systems against adversarial voice commands. They leverage short-term spectral features [19,20] or prosodic features [21, 22] to distinguish different users. However, as the features of human voices typically are low dimensional, advanced passengers can imitate a driver's voice to bypass existing defense systems [23]. Moreover, existing methods on distinguishing different users' identities are generally unreliable in a noisy environment [24], whereas in-vehicle ASR systems demand a much higher security insurance against adversarial voice command attacks to prevent possible car accidents. Furthermore, to prevent malicious voice commands played by speakers, researchers have designed methods to identify whether the voice commands come from humans or loudspeakers. They rely on spectral features [18, 25, 26] and noise features [17, 27] to distinguish between humans and speakers. They are all based on one underlying assumption that voice commands come from a single source. However, in the scenario of driverless cars, malicious voice commands may come from multiple sources (i.e., multiple car speakers). Therefore, the different features for multi-source voices and single-source voices may interfere with distinguishing between human voices and non-human voices [28].

In this paper, we propose a secure automatic speech recognition system called SIEVE to effectively defeat various adversarial voice command attacks on driverless cars. It is capable of distinguishing voice commands issued from a driver, a passenger, and non-human speakers in three steps. First, since legal human voice commands are always single-source signals, SIEVE identifies and filters out multiple-source voice commands from multiple car speakers. The multiple-source detection is based on a key insight that when the same signal is received multiple times in a short time period from multiple sources, the overlap of the received signals will expand the signal correlations in the time domain. Therefore, SIEVE can identify multi-source voice commands by conducting an autocorrelation analysis to measure the overlap of signals.

After filtering out multiple-source voice commands, the second step of SIEVE is to check if a single-source voice command is from a human or a non-human speaker. SIEVE can accurately detect non-human voice commands by checking features in both frequency domain and time domain. First, voices from non-human speakers inherently have the unique acoustic characteristic, i.e., low-frequency energy attenuation. Such characteristic can be checked with the signal power spectrum distribution in the frequency domain. However, sophisticated attackers may use low-frequency enhancement filters to modulate the voice and thus compensate for the energy loss. To identify such modulated voices, SIEVE also conducts a local extrema verification in the time domain. Our key insight is that the local extrema ratio for modulated voices is much greater than that for human voices. Moreover, we demonstrate that attackers cannot modulate voices to bypass the detection in both the time domain and frequency domain at the same time. Hence, our dual-domain check ensures SIEVE can effectively filter out various non-human voice commands.

Finally, SIEVE distinguishes the passenger's voice commands from the driver's voice commands, since we may only trust the driver but not the passengers. Our key insight is that vehicles have fixed internal locations for the driver and passengers. Therefore, we can leverage the directions of voice sources to distinguish the driver's voice and passengers' voice even if passengers can imitate the driver's voice. Particularly, SIEVE measures the directions of voice sources by calculating the time difference of arrivals (TDOA) on a pair of close-coupled microphones (i.e., a dual microphone). To deal with some extreme cases when passengers lean forward and have their head near the headrest of the driver's seat, we also develop a spectrum-assisted detection method to improve the detection accuracy of SIEVE.

We implement a prototype of SIEVE and conduct extensive real-world experiments on a four-seat sedan (Toyota Camry) under various vehicle driving states, including idling, driving on local streets, and driving on highway. The experimental results show that our system can effectively defeat adversarial voice command attacks. For example, SIEVE can achieve a 96.75% accuracy on distinguishing human voices from non-human voices when the car is driving in noisy streets. It can further identify the driver's voice from human voices with a 96.76% accuracy. Moreover, our system can be smoothly integrated to vehicles by replacing the in-vehicle single microphone with a low-priced dual microphone and implanting the detection module in one vehicle electronic control unit.

In summary, we make the following contributions:

- We develop a secure ASR system for driverless vehicles to defeat various in-vehicle adversarial voice commands by distinguishing the command sources from the driver, passengers, and electronic speakers.
- We propose a dual-domain detection method to distinguish voice commands between humans and non-human speakers even if the voices are carefully modulated to mimic human voices.
- We provide a method based on the directions of voice sources to distinguish the driver's voice from passengers' voices even if passengers can imitate the driver's voice.
- We implement a prototype of SIEVE and real-world experimental results show that our system can effectively defeat adversarial voice commands.

2 Threat Model and Assumptions

We focus on the adversarial voice command attacks that manipulate the speech inputs to the in-vehicle ASR system. We assume the vehicle's electronic control unit (ECU) can be trusted [29]. We assume the driver can be trusted to not issue malicious commands; however, malicious voice commands may be issued from in-vehicle loudspeakers, the speakers of mobile devices, or passengers.

First, malicious voice commands may come from the in-vehicle loudspeakers. It is common for people to connect their mobile phones to car audio systems when playing music or making phone calls. Also, CDs/DVDs are usually played through speakers. Since music songs might be downloaded from various untrusted sources, the attackers may edit the sound tracks of audio files to voice the malicious commands via a single speaker or multiple speakers. Particularly, armored attackers may hide adversarial commands by minimizing the difference between the malicious and the original audio samples [7–9]. Moreover, when a phone call is connected to the vehicle speakers via Bluetooth, the people on the other side may unintentionally or intentionally issue voice commands to the ASR systems.

Second, if the driver puts their smartphones on mobile speakers (handsfree mode) when making phone calls or playing musics, a malicious command may be issued from the driver's smartphones. Similarly, a passenger's smartphone may be exploited to voice malicious commands. Thus, it is necessary to identify the voice commands issued from the speakers of mobile devices such as smartphones.

Third, passengers may issue dangerous or annoying human voice commands to ASR systems. For instance, kids

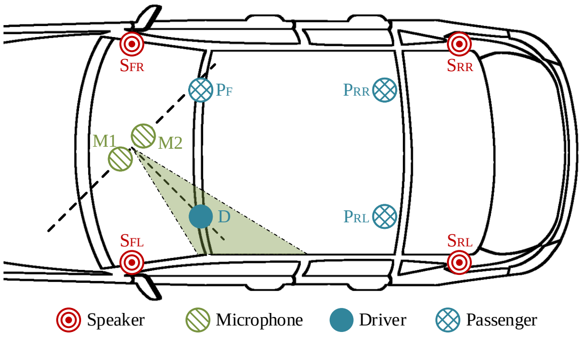


Figure 1: Car Internal Structure and the Location and Orientation of a Dual Microphone.

may unintentionally voice wrong commands to the vehicle or intentionally mess with the recreation systems. Moreover, we assume malicious passengers may bring or leave some dedicated portable hardware to launch advanced attacks such as those in an inaudible frequency range to humans [10–12] (though the sizes of most dedicated hardware devices may not be small). It is critical to distinguish the voice commands from the passengers or their dedicated mobile devices.

3 System Design

We first provide an overview of our system and then present the detailed techniques used in each detection step.

3.1 System Overview

Figure 1 shows the typical internal structure of a four-door sedan, which has at least four speakers installed in four corners (front/rear and left/right) to achieve good stereophonic experience. It has four seats for the driver (D), the front passenger ($\mathrm{P_F}$), the rear left passenger ($\mathrm{P_{RL}}$), and the rear right passenger ($\mathrm{P_{RR}}$).

The entire defense system consists of three detection steps. The first step is to identify and filter out the voices coming from multiple speakers since human's legal commands are issued from single voice source. The second step is to detect adversarial voice attacks from loudspeakers (i.e., replay attacks), no matter the in-vehicle speakers or mobile speakers. The third step is to identify the voice source from its direction by using a dual microphone in the front of the sedan. This step can distinguish voices from the driver and any passenger.

Step 1: Detecting Voice from Multiple Speakers. When attackers use multiple vehicle speakers to perform voice command attacks, the reverberation effect is enlarged since a microphone captures the same signals multiple times at different moments. Since the overlapping of multiple copies of the same signals within a small time expands the signal correlations in the time domain, the linear prediction (LP) residual [30] of the signal can be calculated to decide if the voice commands are received from multiple speakers. Moreover, Hibert envelope and local enhancement techniques are used to enhance the significant excitation. The basic idea is that the relative time delays of the instants of significant excitation remain unchanged in the audio signals captured by the microphones. Therefore, through accumulating the auto-correlation results over the entire voice command signal, we can compare the different patterns to distinguish single-source signals from multi-source signals. Comparing with other methods [28, 31, 32], we adopt the LP residual method since it can achieve a higher detection precision.

Step 2: Distinguishing between Human Voice and Voice from Single Speaker. We develop two new approaches, namely, power spectrum verification and local extrema cross-check, to detect voice from an electronic speaker. Since the common speakers can suppress the power of the low-frequency signals, we use the power spectral density to distinguish the human voice from the replay voice. To escape our power spectrum checking, the attackers may design an inverse filter to compensate the speaker's frequency response. We can defeat this armored attack by performing a local extrema verification in the time domain. By combining these two checks on both frequency-domain features and time-domain features, we can accurately detect the voice coming from loudspeakers.

Step 3: Distinguishing Driver from Passengers. We use a dual microphone to decide the direction of the voice commands. The dual microphone consists of a pair of microphones (M_1 and M_2) that are close to each other (e.g, 5 centimeters). When a voice is captured by the two microphones, we use a far-field model to measure the time difference of arrivals (TDOA) [33] since the source-microphone distance is much larger than the distance between these two microphones. To maximize the detection accuracy, we orient the dual microphone in a direction that the line connecting the two microphones is perpendicular to the line connecting the driver seat and the middle point of the two microphones, as shown in Figure 1. The cross-correlation function of the two-channel signals is effective on measuring the time delay between two channels. As shown in Figure 2, the angle range (inside the vehicle) can be divided into multiple small pie regions, since the TDOA measured value and the propagation angle follow a *arccosine* function (see details in Section 3.4). When the voice propagation direction is perpendicular to these two microphones, the measurement can achieve the highest precision on recognizing the driver. It is the reason why the dual microphone is oriented in Figure 1.

With the orientation of the dual microphone, when a voice comes from the driver's direction, the cross-correlation function is almost central symmetric due to the negligible time delay between two signal channels. When the voice comes from any passenger, the cross-correlation function would be left skewed. In Figure 2, the gray areas represent identification regions for different passengers. In most cases it can accurately distinguish the driver's direction from those of passengers, and we confirm it in our real-world experiments. In some cases, it is challenging to distinguish the driver from

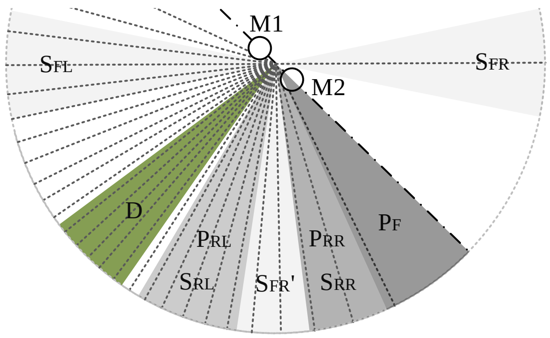


Figure 2: Voice Source Directions to Dual Microphone.

the passenger sitting behind the driver (i.e., $\mathrm{P_{RL}}$), particularly, when the driver may lean towards the right side of the driver's seat (e.g., resting their arms on car armrests) and the passengers may lean forward and have their head near the headrest of the driver's seat. We develop a spectrum-assisted detection technique that combines the location of a particular voice with the voice's specific spectrum features to improve the detection accuracy.

3.2 Detecting Multiple Speakers

The basic idea of the multiple speaker detection is that the reverberation effect occurs since the microphones will capture the same signal multiple times at different instants.

Signal Representations. In the time domain, the captured signals $x(n)=\sum_{i=1}^{m} A_i \cdot s(n-N_i)$ are the overlapping of several time-shifted signals with different attenuation coefficients, where $s(n)$ is the original signal, N_i and A_i denote the time delay and the attenuation coefficient of the i-th signal, and m is the number of speakers. The different time delays depend on the relative locations between microphones and speakers, which are fixed in vehicles. The only difference between the signals captured by two microphones is a slight time shift related to the distance of microphones, so we only need to process the captured signal in any one of the two microphones.

The time delays can be extracted from the significant excitation regions with high signal-noise ratio (SNR). However, the noise will adversely affect the extraction of the time delays. Therefore, it is necessary to reduce the noise and amplify the strong excitations. Also, to reduce the second-order correlations and inhibit the reflection, we extract the linear prediction residuals from the captured signals. According to the parameter estimation model [30], the linear prediction residuals are obtained as $e(n)=x(n)+\sum_{k=1}^{p} a_k \cdot x(n-k)$, where a_k is the predictor coefficients and p is the order of the prediction filter.

Signal Enhancements. The peaks in the linear prediction residuals are of double polarity, which introduces fluctuations to the autocorrelation function. For convenient calculation, linear prediction residuals could be converted into a single polarity form by the Hilbert envelope [34], denoted as $h(n)=\sqrt{e^2(n)+e_h^2(n)}$ where $e_h(n)$ is the Hilbert transform of $e(n)$.

The Hilbert envelop signal can describe the amplitude change of the original signal. However, the weak peaks in the Hilbert envelop may lead to spurious high values in the autocorrelation calculation. Thus, we adopt the local enhancement method to further highlight the high SNR regions. We set a sliding window to calculate the signal mean value in the local area. Then, the Hilbert envelop can be enhanced by taking the square of the original signal over the local signal mean as

$$g(n)=\frac{h^2(n)}{\frac{1}{2M+1}\sum_{k=n-M}^{n+M} h(k)}, \tag{1}$$

where $g(n)$ is a preprocessed signal of Hilbert envelope and $(2M+1)$ denotes the length of the sliding window.

Autocorrelation Analysis. When taking a L-length segment in $g(n)$ as a reference, the autocorrelation function $C(s)$ of the signal $g(n)$ can be calculated as

$$C(s)=\frac{\sum_{k=N}^{N+L-1} g(n)\cdot g(n+s)}{\sqrt{\sum_{k=N}^{N+L-1} g^2(n)\cdot \sum_{k=N+s}^{N+L-1+s} g^2(n)}}, s\in[-S,S], \tag{2}$$

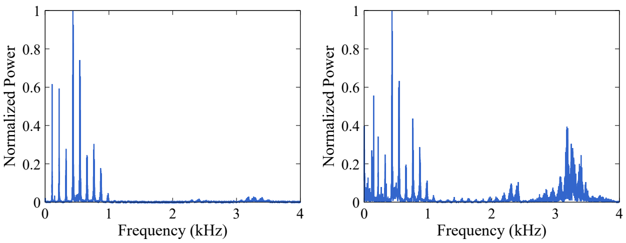
where N is the start index and L is the segment length. The autocorrelation value is normalized by the square mean of the segmented signal. The autocorrelation function is calculated over the interval $[-S,S]$. S indicates the maximum detection range that should be larger than the maximum time delay.

$$S>\frac{D_{max}-D_{min}}{v_0}\cdot f_s=\frac{\Delta D}{v_0}\cdot f_s, \tag{3}$$

where D_{max} and D_{min} indicate the maximum and minimum distance between speakers and microphones, and v_0 is the speed of sound in air with a typical value of 345 *m/s*. f_s is the sampling rate of microphones. Only if S is larger than the maximum possible time delay, the autocorrelation function can record all the time delays information for the speakers.

Judgment Criteria. The autocorrelation function will have several peaks that correspond to the time delays between different propagation paths. For accurate estimation, we judge the results by multiple signal segments rather than a single one. For each autocorrelation function, we only extract the most significant peak that corresponds to the most distinct time delay. In the i-th signal segment, the offset of the highest peak in the autocorrelation function is denoted as $p_i=\arg\max C_i(s)$. To reduce the effects of noise and spurious peaks, we acquire the statistical distribution of p_i with the autocorrelation functions in multiple signal segments.

For a single voice source, most of p_i values are close to zero, resulting in a concentrated distribution. For multiple voice sources, the p_i distribution is rather dispersed. That is because the voices come from different speakers have different arrival moments, resulting in large time delays in the captured signals. Based on these attributes, we can distinguish the patterns between multiple-speaker signals and single-speaker signals according to the dispersion of the p_i distribution. The dispersion P is measured as the proportion of p_i in the interval



(a) The human voice

(b) The replay voice

Figure 3: Power Spectral Density for Voice Signals.

$[-d,d]$, where d is a small number compared with S. λ is the decision threshold that obtained from multiple experiments. If $P \geq \lambda$, it means the time delays are centered around 0 and the voice comes from a single source. If $P < \lambda$, the voice comes from multiple sources due to the dispersed distribution. Since the voices from multiple sources are most likely adversarial voices generated by attackers, we can safely filter out these multi-source voices in the first step.

3.3 Identifying Human Voice

After verifying a voice coming from a single source, we detect and filter out the voice that comes from loudspeakers. We solve this challenge by combining two approaches, namely, *frequency-domain power spectrum verification* and *time-domain local extrema cross-check*, to ensure that the voice indeed comes from humans.

Frequency Domain Verification. This approach is based on a noticeable timbre difference between a human voice and a replay voice sound from loudspeakers. Human beings voice commands through the phonatory organ, resulting in a sound frequency typically from 85 Hz to 4 kHz [35]. However, a dynamic loudspeaker can suppress the signals in the low-frequency range due to the limited size, especially under the frequency of 500 Hz [36]. Thus, even a speaker replays a recorded human voice that contains the same frequency components, the timbre is totally different from the genuine one. The main reason is that different power distributions of frequency components lead to different timbre [37].

By leveraging the characteristic of different power distributions, we can distinguish the voice coming from a human or a loudspeaker. The captured voice will be verified with the power spectral density, specifically the ratio of the low-frequency power. The frequency of human voice ranges from 85 Hz to 4 kHz, among which the low-frequency components are dominant, as shown in Figure 3(a). The replay audio sound from loudspeaker has the similar frequency components; however, the sharp decrease in the low-frequency components increases the relative ratio of the high-frequency components, as shown in Figure 3(b). In our design, a voice that comes from a single source is further verified by evaluating the power ratio of the low-frequency components (85 Hz - 2 kHz) to all frequency components. If the voice indeed comes from humans, the power ratio should be greater than a specific threshold.

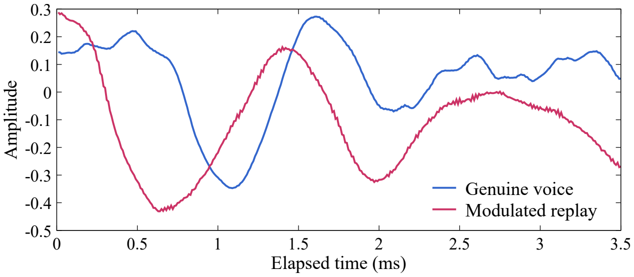


Figure 4: Time-domain Waveform for Voice Signals.

However, an ingenious attacker may compensate the loss of the low-frequency energy by modifying the recording file in the frequency domain. By estimating the transmission properties $K(f)$ of the loudspeakers, an attacker can design an inverse filter $K^{-1}(f)$, where $K(\hat{f}) \cdot K^{-1}(f) = 1$. Then the attacker can reconstruct the audio file with $K^{-1}(f)$ for compensating the speaker frequency response, and we call such voice as *modulated replay voice*. As the frequency response of the loudspeaker and the inverse filter cancel each other during playback, it is difficult to solely rely on the frequency-based method for distinguishing the modulated replay voice from the genuine voice. Fortunately, we could combine a verification approach in the time domain.

Time Domain Verification. We observe there are different patterns in the local extrema ratio of the human voice and the modulated replay voice. In a 3-length window of time-domain signal, if the midpoint is the maxima or minima in the window, we define the midpoint as a *local extrema* [38]. Also, the ratio of the local extrema amount to the total signal length is defined as *local extrema ratio*. Though the local extrema are not directly related to the spectrum, the number of local extrema can indirectly reflect some spectrum features.

The attacker can only compensate the voice signals with the amplitude spectrum. The phase spectrum is hard to be compensated because of the difficulty to measure the speaker phase response. Due to the phase mismatch errors in the modulated voice, the time-domain signal will contain extremely small oscillation, namely ringing artifacts (see Figure 4). These artifacts cannot be heard by a human, but the local extrema ratio of the modulated replay voice is much greater than that of the human voice in the time domain.

Because the local extrema ratios of the human voice and the modulated replay voice are different, we can identify if the voice indeed comes from a human or a loudspeaker by combining both the frequency-domain power spectrum verification and the time-domain local extrema cross-check. A verified human voice command must satisfy two conditions: (1) the low-frequency power dominates; (2) it complies with the human voice patterns in local extrema ratio. It is difficult for attackers to meet both requirements by manipulating the limited-sized loudspeakers.

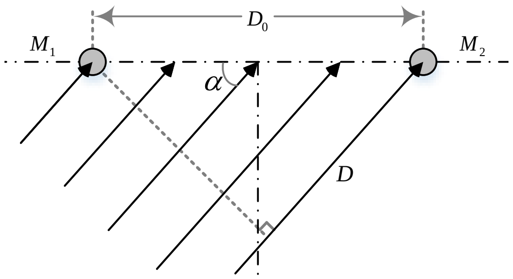


Figure 5: Principle of Time Difference of Arrival.

3.4 Identifying Driver's Voice

In the third step, we identify the voice source via the voice propagation direction and thus distinguish the commands voiced from the driver or any passenger.

Time Difference of Arrival. Two spatially separated microphones can detect the voices from different propagation directions using the time difference of arrival (TDOA), as shown in Figure 5. The distance between M_1 and M_2 is denoted as D_0. Since the distance between voice source and microphones is larger than the distance between two microphones, we use a far-field model [39] to calculate the time difference of arrivals. The angle between the voice propagation direction and M_1M_2 is denoted as α. If α is 0 or π, the propagation direction is parallel to M_1M_2, resulting in the largest difference of arrival. If α is equal to $\pi/2$, the propagation direction is perpendicular to M_1M_2, and the difference of arrival will be zero. Due to the microphone placement shown in Figure 1, zero difference of arrival means the voice comes from the driver.

In Figure 5, D is the difference of the propagation distances between the voice source and two microphones. Thus, the time difference of arrivals can be calculated as $\Delta t = D/v_0 = (D_0 \cdot \cos\alpha)/v_0$. Because the voice is recorded as a digital signal, the captured voice is discrete in the time domain. In the captured signal, the difference of arrivals in sampling units ΔN can be estimated as $N_0 - 1 < \Delta N \le N_0 + 1$, where $N_0 = \Delta t \cdot f_s$. To simplify the calculation, we approximate ΔN as $\Delta N \propto D$. Thus, we can obtain $\Delta N \approx D_0 \cdot \cos\alpha \cdot f_s/v_0$. The propagation angle α that calculated as follows will be used to determine the direction of the voice source.

$$\alpha = \arccos(\frac{\Delta N \cdot v_0}{D_0 \cdot f_s}). \tag{4}$$

Detection Precision. The effects of the ΔN changes on α are different. When α is approximately equal to $\pi/2$, for each change of one unit in ΔN, the change in α can be calculated according to Taylor series expansion: $\Delta\alpha = x + o(x)$, where $x = v_0/(D_0 \cdot f_s) < 1$. When α approaches 0 or π, for each change of one unit in ΔN, the change in α can be calculated according to Puiseux series expansion: $\Delta\alpha = \sqrt{2x} + o(x)$.

As a result, if $\alpha \approx \pi/2$, for each unit change of ΔN, $\Delta\alpha$ will be less than that one when $\alpha \approx 0$ or π. More units are concentrated around $\alpha \approx \pi/2$. Thus, the system becomes more sensitive to the angle change near the driver's direction, providing the optimal detection precision for the driver's voice.

Signal Preprocessing. When a sound signal arrives, we first check if it is a usable signal. Since the frequency of speech signal is between 85 Hz (f_l) and 4000 Hz (f_h), a fast Fourier transform (FFT) is used to obtain the frequency spectrum $X(k)$ of the captured signal $x(n)$, where $X(k) = FFT(x(n))$. Then we judge the speech signal by verifying the power ratio of a band-pass signal $Rp = \sum_{k=k_l}^{k_h} X^2(k) / \sum_{k=0}^{K/2} X^2(k) > \varepsilon$. K is the amount of points in the FFT. $k_l = \lfloor K f_l / f_s \rfloor$, $k_h = \lfloor K f_h / f_s \rfloor$, where $\lfloor x \rfloor$ means the largest number less than x. $\varepsilon = 0.57$ is a threshold value obtained from our experiments. The signal will only be processed in this step when $Rp > \varepsilon$, because the higher SNR signal is suitable for the TDOA algorithm [40].

To reduce high-frequency noise and obtain a smooth signal waveform, we process the captured signal with a pre-set low-pass filter in the frequency domain. The smooth voice signal $y(n)$ can be obtained by the inverse fast Fourier transform (IFFT). $y(n) = IFFT(X(k) \cdot H(k))$, where $H(k)$ is a low pass filter that inhibits the high frequency components.

Cross-Correlation Evaluation. According to the TDOA algorithm, the cross-correlation function between two-channel signals is given by the following equation.

$$C_{12}(s) = \sum_{n=n_0}^{n_0+l-1} y_1(n-s) \cdot y_2(n), -S_m \le s \le S_m, \tag{5}$$

where n_0 is the start index, l is the segment length, $y_1(n)$ and $y_2(n)$ are the signals captured by M_1 and M_2. S_m is the max shift value that subjects to the constraint $S_m \ge D_0 \cdot f_s / v_0$.

In the cross-correlation function, the shifted sampling unit with the maximum cross-correlation value indicates the time delay between two channels. The corresponding offset value of the cross-correlation peak is denoted as $s_0 = \text{argmax}(C_{12}(s))$. According to Equation (4), the voice propagation angle can be estimated as $\alpha = \arccos[(s_0 \cdot v_0)/(D_0 \cdot f_s)]$. Note if a voice comes from any passenger, s_0 will be a negative value as the propagation angle α is greater than $\pi/2$.

In Figure 1, the decision criterion is $|\alpha - \pi/2| \le \alpha_T$ if a signal is recognized as the driver's voice. α_T is an angle threshold that demarcates the decision boundary. Considering the relationship between α and s_0, we only need to use the decision criterion $|s_0| \le s_T$ with an offset threshold s_T. If s_0 satisfies the above condition, it means the voice comes from the direction approximately perpendicular to M_1M_2. Thus, the captured voice can be recognized as coming from the driver.

Spectrum-assisted Detection. In real-world situation, the driver may lean to the right side on the armrest during driving, and its voice may fall into the angle range of the rear-left passenger. To identify the driver's voice more robustly, we develop a spectrum-assisted detection technique to allow the voice of the driver to move within a wider angle range without sacrificing the detection accuracy. The basic idea is to combine specific spectrum characteristics of the wake-up voice command (e.g., "Hi, SIEVE") with the direction of the voice.

To determine the same voice source (i.e., the driver), we record the spectrum histogram and the propagation direction of previous wake-up commands. For the i-th command that has been successfully recognized as the driver's voice, the m-bar spectrum histogram of the wake-up command is denoted as $v_j^i,\ j = 1, \ldots, m$, and the propagation angle for the i-th voice command is denoted as α^i. For the next $(i+1)$-th command, the received wake-up command must satisfy two conditions. First, the spectrum statistics of wake-up commands are similar, indicating the voice commands come from the same person. The spectrum similarity can be measured using the root-sum-square of histogram difference $(\sum_j (v_j^{i+1} - v_j^i)^2)^{(1/2)} < th_1$, where th_1 is a similarity threshold. Second, the voice movement is within an acceptable wider range (e.g., the driver's voice cannot come from the seats on the right), which is measured by the angle difference $|\alpha^{i+1} - \alpha^0| < th_2$, where $\alpha^0 = \pi/2$ is the theoretical measured angle of the driver. And the angle threshold th_2 can be $\pi/4$, indicating that the driver sits on the left side of the car. If a newly received wake-up command satisfies both conditions, we would consider the voice command is coming from the driver.

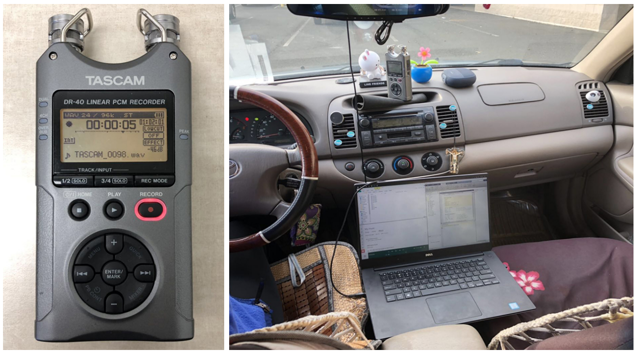
With the spectrum-assisted detection method, our system can successfully recognize the driver's commands even though the driver is in a different sitting posture. Also, if the person sitting in the seat directly behind the driver leans forward and has his head near the headrest of the driver's seat, our system can still reject his commands since the commands only satisfy the angle constraint but not both constraints.

4 Experimental Results

In our experiments, a TASCAM DR-40 portable digital recorder with two spatially separated microphones is used to capture the voice signals, as shown in Figure 6(a). The distance between the two receivers D_0 is 5 *cm*. The sampling rate of both microphones is 96 kHz (f_s). We not only analyze the data captured in the lab environments, but also test our system in real world, as shown in Figure 6(b). The vehicle model is Toyota Camry LE 06 with two Scion TC XB 6.5-inch speakers and two Kicker 43DSC69304 D-Series 6x9-inch speakers. To test more loudspeakers, we use three smartphones (iPhone X, Google Nexus 5, and Xiaomi Mi 4) with their built-in speakers. Since we cannot modify the electronic control unit (ECU) of the vehicle, the system runs in an environment on a laptop with Intel Core i7-7700, 2.8GHz CPU with 16GB RAM.

4.1 Accuracy on Detecting Multiple Speakers

In the multiple speakers detection, the linear prediction residuals are obtained by a 12-order linear prediction filter. A sliding window with 2001 units length is used to enhance the preprocessed signals. The length of the signal segments is 512 units, and the maximum offset value is also 512 units because the maximum distance difference between speakers and microphones is 1.5 meter in our experiments. According to Equation (3), S should be greater than 417 units. The threshold λ is set as 0.33 through 42 experiments. Under this condition, the total recognition accuracy can reach 83.3%.



(a) Tascam DR-40. (b) Vehicle Testbed.

Figure 6: Experiment Setup.

We conduct experiments by using different combinations of speakers. The test audio was originally collected by the digital recorder with a sampling rate of 96 kHz. One 4-track audio file with 15 track combinations (4 combinations for 1 speaker, 6 combinations for 2 speakers, 4 combinations for 3 speakers, and 1 combination for 4 speakers) is edited via MATLAB vector operations. The test recording file is finally generated by the *wavwrite* tool [41]. The detection accuracy of a single or four in-vehicle speakers is 100%, while the average accuracy of detecting two and three speakers is 66.7% and 75%, respectively. It is challenging to identify two front speakers or two rear speakers since in that case ΔD is small and easy to ignore. However, when considering the voice from those two speakers as from one source, we still can filter them out according to their directions in the third step.

4.2 Accuracy on Detecting Human Voice

By evaluating the power ratio of low-frequency components, we can distinguish the human voice from the replay voice sound from loudspeakers. In Figure 7(a), 97.3% of human voices have the low-frequency power ratio of over 0.995. The low-frequency power ratio of a replay voice is distributed and less than that of a human voice. Based on these features, we can distinguish if the voice commands sound from the driver or a loudspeaker. In our experiments, the detection accuracy on replay voices is 99.05% with the threshold of 0.96.

We also confirm that the low-frequency power of the modulated replay voice dominates after the artificial enhancement, which makes our frequency-based method unreliable. Therefore, we should also cross-check the voice in the time domain, where the voice pattern can be obtained by calculating the local extrema ratio. The pattern difference of the human voice and the modulated replay voice is illustrated in Figure 7(b). Because of the ringing artifacts, the modulated replay voice has a larger local extrema ratio, typically greater than 35%.

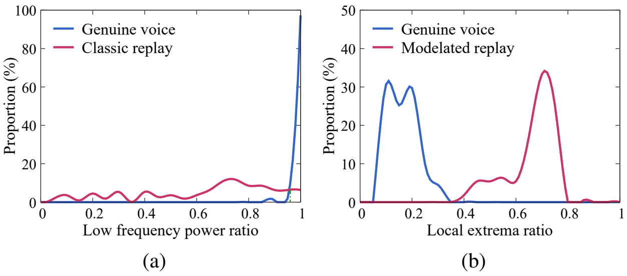


Figure 7: Frequency-domain and Time-domain Features for Detecting Human Voice. (a) the low-frequency power ratios between human voice and replay voice; (b) the local extrema ratios between human voice and modulated replay voice.

While the local extrema ratio of human voice is typically less than 35% due to the statistical smooth.

Therefore, we can successfully distinguish the human voice from the modulated replay voice via the local extrema cross-check methods. When the decision threshold is 0.35, the detection accuracy can achieve 99.62%.

4.3 Accuracy on Detecting Driver's Voice

A series of experiments are conducted to distinguish voices coming from the driver or a passenger. The length of the signal segments is 512 units. The maximum offset S_m is 64, which should be greater than the theoretical maximum peak offset $(\Delta N)_{max} \approx 14$ units.

We first perform experiments with five basic voice propagation angles (0, $\pi/4$, $\pi/2$, $3\pi/4$, π) in a quiet lab environment. Figure 8(a) shows the results of cross-correlation in logarithmic form with normalization, verifying the correctness of our theoretical analysis. When the propagation angle α is 0 or π, the absolute offset of the peak $|s_0|$ is 14 units, the same as $(\Delta N)_{max}$. When α is $\pi/4$ or $3\pi/4$, the absolute offset of the peak $|s_0|$ is 10 units, which follows the equation $s_0 \approx (\Delta N)_{max} \cdot cos(\alpha)$. The peak offset s_0 is near 0 when the voice comes from the direction perpendicular to M_1M_2. This property enables the system to distinguish the driver's voice and the voice coming from any passenger. Another intriguing property in Figure 8(a) is the different detection precision over various angles. The α changes from 0 to $\pi/4$ are presented by 4 units, while 10 units are used to measure the α changes from $\pi/4$ to $\pi/2$. Higher precision can be achieved near the angle of $\pi/2$, which means we can get the best detection performance in the driver's direction. It explains why we orient two microphones with a 45-degree angle to the vehicle.

We also conduct experiments in a real car to distinguish the driver's voice from passengers' voices. Figure 8(b) shows the cross-correlation functions of the driver's voice and three passengers' voices. The cross-correlation functions of the passengers are left-skewed with negative peak offsets since the propagation angles are greater than $\pi/2$. The driver's voice has a near-zero peak offset due to the propagation angle of $\pi/2$. The absolute peak offset of P$_{\text{Rear-L}}$ is 6 units, which is the closest to the driver. Thus, according to the decision criterion, SIEVE can distinguish the driver from three passengers with a decision threshold (s_T) of 2. In the experiments, 2437 signal segments are tested and the overall detection accuracy can achieve 96.76%. The false positive rate is 2.86%, while the false negative rate is 4.44%. The statistics show the system is accurate and robust when distinguishing the driver's voice from the passengers' voices.

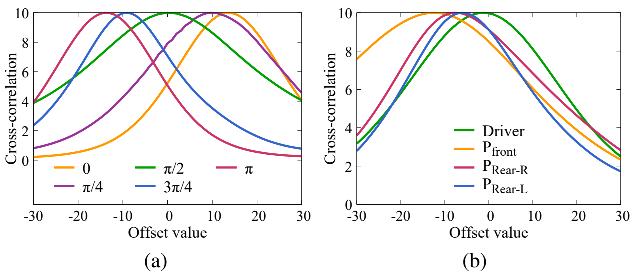


Figure 8: Experimental Results under Different Situations. (a) cross-correlation functions for 5 propagation angles in a quiet lab environment; (b) cross-correlation functions for the driver and three passengers.

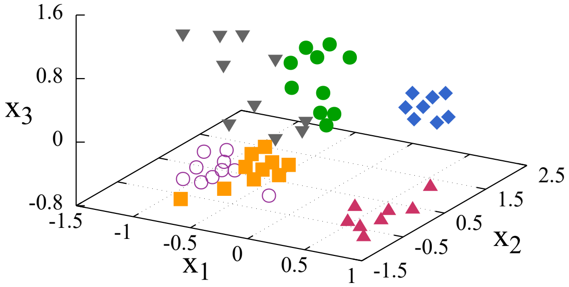


Figure 9: The Spectrum Features of the Wake-up Command for 6 Users in PCA Low-dimensional Subspace.

To verify the validity of the spectrum-assisted detection method, we collect 60 wake-up commands that are issued from 6 different users. Then we utilize a 10-bar spectrum histogram (from 0 to 2 kHz) to extract the spectrum features of the wake-up command. With the similarity threshold th_1 of 0.081, the accuracy of correlating two voice commands can achieve 92.72% (i.e., only 262 out of 3600 pairs are misjudged). After applying the PCA dimension reduction [42], we can visualize the spectrum features in a 3-D subspace shown in Figure 9. The features of a single user form a cluster, which clearly differs from other clusters that represent different users. The PCA subspace verifies the effectiveness of the spectrum-assisted method. Moreover, it only takes 19 *ms* to check the spectrum similarity constraints.

4.4 System Robustness

We conduct extensive experiments to study the system robustness, which may be impacted by vehicle driving states, the

# of Speakers	Idling	Local	Highway
1	100%	83.3%	58.3%
2	66.7%	58.3%	66.7%
3	75%	66.7%	75%
4	100%	100%	100%
Total	83.3%	73.8%	71.4%

Table 1: The Detection Accuracy for Different Number of Speakers Under Different Driving States.

placement of microphones, and driver's sitting positions.

Vehicle Driving States. Three types of driving states are tested in our experiments: *idling*, *driving on local streets*, and *driving on highway*. Idling refers to running a vehicle's engine but the vehicle is not in motion. The car is in a low-noise environment during idling. Driving on local streets means that the car runs at a low speed of around 20 miles per hour, where the car is usually in a medium-noise environment. Driving on highway indicates that the car runs at about 50 miles per hour on highway, with the highest level of environmental noise.

First, when detecting multiple speakers under each driving conditions, 42 voice segments are collected as the inputs of the autocorrelation algorithm to judge if the sample comes from multiple speakers. The experimental results for multiple speakers detection are shown in Table 1. We can see that the detection accuracy decreases gradually from the idling condition to the highway condition. Among them, the most significant change is the single speaker detection accuracy, which decreases considerably with different conditions. When driving on highway, the outside noise is so complex and unpredictable that the received signals contain a lot of noise peaks, which generate spurious high values in the autocorrelation calculation. Therefore, some signals from a single speaker may be incorrectly classified as multiple-source signals. The problem can be solved by using sound absorption material, better denoising algorithm, or multiple microphones scheme.

Second, we evaluate the detection accuracy of human voice under three driving states. Table 2 shows the driving states have little impact on the human voice detection, since the signal power is much greater than the noise power. Compared with the idling state, the detection accuracy only decreases by 3.26% when driving on the highway. Also, we discover that the driving noise has a higher influence on the time-domain verification than the frequency-domain verification, because the driving noise mainly affects the waveform in the time domain, not the statistical values in the frequency domain.

Third, one big challenge for the driver's voice identification is the interference from outside noise. When the received signals are mixed with strong noise, there will be unexpected fluctuations in the cross-correlation function. These fluctuations will eventually offset the expected peaks, usually in the 0-offset direction due to the common-mode interference [43]. Table 3 shows the results of distinguishing the driver and the passengers. In the case of high interference, the driver's voice can still be distinguished from the passengers' voice but the offset discrimination becomes moderate.

Driving State	Accuracy
Idling	97.46%
Driving on Local Street	96.75%
Driving on Highway	94.20%

Table 2: The Detection Accuracy of Human Voice under Different Driving States.

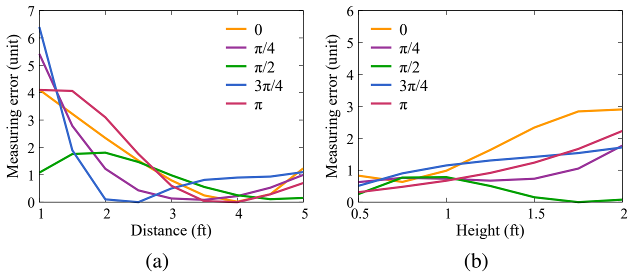
Voice Source		Idling	Local	Highway
Driver	Mean	-0.11	0.38	1.09
	Stdev	4.15	3.03	2.11
Front Passenger	Mean	-11.31	-10.99	-8.88
	Stdev	5.98	4.67	4.75
Rear Right Passenger	Mean	-8.02	-6.57	-5.31
	Stdev	4.04	3.29	5.00
Rear Left Passenger	Mean	-5.36	-5.30	-4.57
	Stdev	3.58	3.27	3.75

Table 3: The Peak Offsets for the Driver and Passengers under Different Driving States.

Relative Distance and Height from Microphones. In our scheme, the TDOA model is based on a far-field model.A quantitative experiment is conducted to evaluate the impacts of the distance between the voice source and the microphones and the height of microphones relative to the horizontal plane.

To evaluate the impacts of the voice source distance, 25 experiments are conducted and 2728 sample segments are acquired. In our experiments, 5 propagation angles (0, $\pi/4$, $\pi/2$, $3\pi/4$, π) are tested. The testing range for voice source distance is from 1 foot to 5 feet with a spacing of 1 foot. From the experimental results illustrated in Figure 10(a), we can see that the measurement error is less than 2 offset units when the voice source distance is greater than or equal to 3 feet. Also, when the distance is less than or equal to 2 feet, the measurement error increases since the assumption of the far-field model is not applicable. However, inside vehicles, most sound sources are more than 2 feet away from the microphones.

To explore the effect of the voice source relative height on measurement accuracy, we conduct 20 experiments and collect 1795 voice segments. We evaluate the measurement error in 5 different propagation angles (0, $\pi/4$, $\pi/2$, $3\pi/4$, π). Because of the far-field model, the horizontal distance between the microphones and the testing voice source is set to 3 feet, and the relative vertical height of the testing voice source is set to 0.5, 1, 1.5, and 2 feet, respectively. The experimental results are shown in Figure 10(b). With the propagation angle of $\pi/2$, the voice source relative height has little impact on the measuring accuracy, since the relative height does not introduce an additional distance difference. However, when the propagation angle is not $\pi/2$, the measuring error increases



(a)

(b)

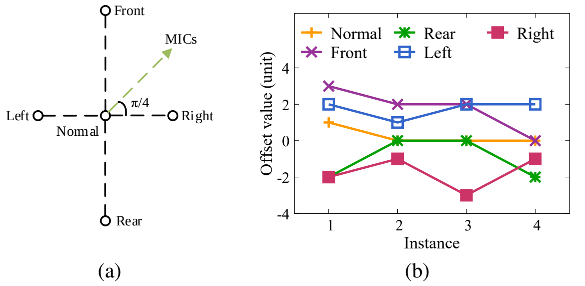
Figure 10: Measurement Accuracy on Distance and Height. (a) measurement error vs. relative distance; (b) measurement error vs. relative height.

with the relative height of the voice source because of the extra distance differences of arrival. The measuring error is more obvious when the propagation angle is close to 0 or π. However, the measuring errors in all cases are less than 3 offset units when the relative height of the voice source is not larger than 2 feet. Therefore, the relative vertical height of the voice source has minimal impact on our measurement.

Driver's Sitting Positions. As the driver's seat can be adjusted according to the driver's preferences, the driver's voice source would move forward or backward. In addition, according to different drivers' driving habits, the driver's position may lean towards left or right side. Therefore, we need to explore the influence of the different sitting positions on the measurement accuracy. As shown in Figure 11(a), the five most common sitting positions (normal, front, rear, left, right) are used to test the measurement accuracy. The microphones are positioned at a 45 degree angle to the front right of the driver's normal position. For each sitting position, 4 sample instances are tested, and the experimental results are shown in Figure 11(b). In ideal circumstances, a voice signal comes from a normal position will have a zero offset value in the cross-correlation function. However, in the real-world situation, the measurement error is 1 offset value for the voice coming from the normal position. When the driver moves forward or left, the voice propagation angle will be slightly less than π/4, and the voice cross-correlation will have a positive peak offset. When the driver moves back or right, the voice propagation angle will be a little greater than π/4, and the voice cross-correlation will have a negative peak offset. The offset values of the front position are larger than those of other positions, while the offset values in the right position are smaller than others. As the absolute offset values in most cases are less than 3, different sitting positions have little impact on the detection results. Thus, the decision threshold s_T is set to be 2, which can successfully distinguish the driver's voice even at different sitting positions.

4.5 Performance Overhead

Table 4 shows the the system performance overhead including running time and memory size for each detection step. Running time is measured as the average processing time for a single voice sample, and memory overhead measures the memory space occupied by the running program. The total running time for a voice sample is 214 ms in single-core mode with a 2.8GHz CPU, and the total occupied memory size is 144 MB. Since the running time is measured using slow Matlab code, we believe the optimized C code or assembly code may further reduce the running time.



(a)

(b)

Figure 11: Experimental Results of Measurement Accuracy on Different Sitting Positions. (a) relative position of different sitting positions. (b) offset values of the instances with different sitting positions.

Detection Step	Running Time	Memory
Multi-speaker Detection	134 ms	111 MB
Human Voice Detection	47 ms	10 MB
Driver's Voice Identification	33 ms	23 MB
Total Overhead Costs	214 ms	144 MB

Table 4: Performance Overhead for Detection Step.

Our system can be well supported by the modern in-vehicle computing platforms. For example, the in-vehicle embedded computing devices by Neousys Technology are configured with 2.3GHz-3.6GHz processor, up to 64GB DDR4 RAM [44]. FPGA-based or GPU-based hybrid Electronic Control Unit (ECU) can achieve hardware acceleration for in-vehicle computing [45] [46]. Since the driverless vehicle can deploy multiple ECU modules for different tasks [47], it is feasible to embed our system into a dedicated ECU module to avoid interfering other tasks running on other ECUs.

5 Discussions

Though our system design is customized to one popular sedan internal structure, it can be extended to other vehicle models or future driverless car models. In the left-driving countries (e.g., U.K. and Japan) where the driver's seat is on the right, it is easy to adopt our system in those cars by mirroring the placement of the microphones and adjusting the location algorithm accordingly. Though most cars have four built-in speakers installed at four window corners, some models of cars have different numbers of speakers installed at different locations [48]. Since our speaker detection mechanism is effective on detecting the voice coming from more than one speaker, it works well on different number of speakers.

Our current system design uses as few as two microphones close to each other. Thus, it is convenient to be installed as either built-in or external car microphone system with minor change on vehicle's interior design. The microphones in our prototype can divide the angle change between 0 and π into 28 regions; however, the microphones with a higher sampling rate may provide a fine-grained angle measurement. Also, a high-end microphone can reduce the noise in the background by supporting advanced denoising algorithms such as Fourier Bessel expansion [49]. It is also plausible to deploy more microphones (or a microphone array) in the future car designs. Thus, we can achieve a more accurate localization of the sound source in the three-dimensional space [50].

Our system integrates several detection methods that can be generalized and applied in other applications. For instance, the multiple speaker detection technique may be adopted in smart home systems to prevent malicious voice commands from household speakers. In the circumstances where sound sources have relatively fixed locations, our single voice source identification solution may be useful to determine the identity of the source. Moreover, our replay voice detection solution can be applied to enhance the security of voice-activated smart doors or other IoT devices.

6 Related Work

Automatic Speech Recognition (ASR) Systems. An automatic speech recognition system converts the speech signal into recognized words, which could be the inputs to natural language processing. Since it requires little special training and leaves hands and eyes free, ASR is ideal for drivers to issue commands to the vehicle systems. According to the capabilities of ASR systems, an ASR system can support either isolated word or continuous speech, read speech or spontaneous speech, speaker-dependent or speaker-independent [51], small vocabulary or large vocabulary, finite-state or context-sensitive language model [52], and high or low SNR [53].

Attacks on ASR Systems. The ASR systems are vulnerable to several voice-based attacks. For existing speaker identification solutions [54, 55], it remain as a challenge to defeat armored impersonation attacks [23, 56] and replay attacks [17, 18, 26]. Speech synthesis attack [57, 58] is a relatively complex method to perform attacks by a text-to-speech conversion. With the development of adversarial learning, more sophisticated attacks have emerged. Dolphin Attack [10, 59] utilizes ultrasonic modulation to move voice commands to an undetectable frequency band. Psychoacoustic model can be leveraged to generate the adversarial voices below the human perception threshold [6]. Voice commands can also be injected into voice controlled devices by laser modulation [60]. Some malicious voice commands can be understood by ASR systems, but not by humans [61]. Thus, attackers can hide voice commands in noise-like signals and control the mobile voice recognition systems [8]. CommanderSong [9] demonstrates a more practical attack that embeds voice commands into music songs without being noticed by human beings.

Attacks on NLP module. Threats may also come from the natural language processing module of the voice control systems. Zhang et al. focus on the intent classifier in NLP module, generating semantic inconsistency by specific interpretation [62]. Moreover, an attack called *Skill Squatting* utilizes systematic error to hijack voice commands on Amazon Alexa and route them to a malicious third-party application with a similarly pronounced name [63]. Mitev et al. use skill-based Man-in-the-Middle attack to hijack conservation between Alexa and victims [64]. Attackers can also leverage voice masquerading attack to steal users information by impersonating as a legal application and communicating with the users [65].

Sound Source Localization. The most popular way of sound source localization is to utilize the time delays of arrival (TDOA) in different sound receivers [66]. Particularly, the direction-of-arrival (DOA) can be measured on a pair of microphones [67]. DOA techniques can be used to verify voice commands for IoT devices [68]. DOA is also utilized to detect articulator dynamic within a short distance for securing the mobile ASR systems [69, 70]. The 3-D sound source can be determined by using an array of multiple microphones [71,72]. With different installations of microphone array, such as planar array [73] or rectangular prism [39], we may use different location estimation algorithms. An advanced method uses blind source separation technique to locate multiple sound sources simultaneously [74]. Moreover, sound source localization can also be achieved by other methods such as Gaussian mixture models [75] or golden section searching [76].

7 Conclusion

In this paper, we propose a secure in-vehicle ASR system called SIEVE to defeat adversarial voice command attacks on voice-controlled vehicles. We utilize the physical attributes of voices to distinguish the driver's voice from other adversarial voices in three steps. First, multi-source signals are filtered out according to the diffusion of autocorrelation on linear prediction residuals. Second, voice attacks from non-human speakers are filtered out by cross-checking both the frequency domain and time domain. Third, the driver's voice is determined from its propagation direction with a dual microphone. We implement a system prototype and conduct experiments in real cars. The experimental results show our system can achieve a high detection accuracy in real-world situations.

Acknowledgments

This work is partially supported by the U.S. ARO grant W911NF-17-1-0447, U.S. ONR grant N00014-18-2893, and the NSFC grant 61572278. Jiahao Cao and Qi Li are the corresponding authors of this paper.

References

[1] Lex Fridman, Daniel E. Brown, Michael Glazer, William Angell, Spencer Dodd, Benedikt Jenik, Jack Terwilliger, Julia Kindelsberger, Li Ding, Sean Seaman, Hillary Abraham, Alea Mehler, Andrew Sipperley, Anthony Pettinato, Bobbie Seppelt, Linda Angell, Bruce Mehler, and Bryan Reimer. MIT autonomous vehicle technology study: Large-scale deep learning based analysis of driver behavior and interaction with automation. *CoRR*, abs/1711.06976, 2017.

[2] Tesla Autopilot. Wikipedia, the free encyclopedia, 2018. [accessed 20-November-2018].

[3] Waymo. Wikipedia, the free encyclopedia, 2018. [accessed 20-November-2018].

[4] Carplay. Wikipedia, the free encyclopedia, 2019. https://en.wikipedia.org/wiki/Carplay, [accessed December 2019].

[5] Automated driving system. Wikipedia, the free encyclopedia, 2019. https://en.wikipedia.org/wiki/Automated_driving_system, [accessed December 2019].

[6] Lea Schönherr, Katharina Kohls, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. *CoRR*, abs/1808.05665, 2018.

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

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

[9] 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. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 49–64, Baltimore, MD, 2018. USENIX Association.

[10] Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. Dolphinattack: Inaudible voice commands. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, pages 103–117, 2017.

[11] Nirupam Roy, Sheng Shen, Haitham Hassanieh, and Romit Roy Choudhury. Inaudible voice commands: The long-range attack and defense. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, pages 547–560, Renton, WA, 2018. USENIX Association.

[12] Liwei Song and Prateek Mittal. Inaudible voice commands. *CoRR*, abs/1708.07238, 2017.

[13] Yuan Gong and Christian Poellabauer. An overview of vulnerabilities of voice controlled systems. *CoRR*, abs/1803.09156, 2018.

[14] V. L. L. Thing and J. Wu. Autonomous vehicle security: A taxonomy of attacks and defences. In *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 164–170, Dec 2016.

[15] Wonkyum Lee, Kyu J. Han, and Ian Lane. Semi-supervised speaker adaptation for in-vehicle speech recognition with deep neural networks. In *Interspeech 2016*, pages 3843–3847, 2016.

[16] Rosa González Hautamäki, Tomi Kinnunen, Ville Hautamäki, Timo Leino, and Anne-Maria Laukkanen. I-vectors meet imitators: on vulnerability of speaker verification systems against voice mimicry. In *INTERSPEECH*, 2013.

[17] Z. Wang, G. Wei, and Q. He. Channel pattern noise based playback attack detection algorithm for speaker recognition. In *2011 International Conference on Machine Learning and Cybernetics*, volume 4, pages 1708–1713, July 2011.

[18] J. Villalba and E. Lleida. Preventing replay attacks on speaker verification systems. In *2011 Carnahan Conference on Security Technology*, pages 1–8, Oct 2011.

[19] V. Hautamäki, T. Kinnunen, F. Sedlák, K. A. Lee, B. Ma, and H. Li. Sparse classifier fusion for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(8):1622–1631, Aug 2013.

[20] Nicholas Evans, Tomi Kinnunen, Junichi Yamagishi, Zhizheng Wu, Federico Alegre, and Phillip De Leon. *Speaker Recognition Anti-spoofing*, pages 125–146. Springer London, London, 2014.

[21] A. G. Adami, R. Mihaescu, D. A. Reynolds, and J. J. Godfrey. Modeling prosodic dynamics for speaker recognition. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, volume 4, pages IV–788, April 2003.

[22] L. Ferrer, N. Scheffer, and E. Shriberg. A comparison of approaches for modeling prosodic features in speaker recognition. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4414–4417, March 2010.

[23] Johnny Mariéthoz and Samy Bengio. Can a professional imitator fool a gmm-based speaker verification system? Idiap-RR Idiap-RR-61-2005, IDIAP, 2005.

[24] David Gerhard. Pitch extraction and fundamental frequency: History and current techniques. Technical report, 2003.

[25] Marcin Witkowski, Stanislaw Kacprzak, Piotr Zelasko, Konrad Kowalczyk, and Jakub Galka. Audio replay attack detection using high-frequency features. In *INTERSPEECH 2017*, 2017.

[26] Tomi Kinnunen, Md Sahidullah, Héctor Delgado, Massimiliano Todisco, Nicholas Evans, Junichi Yamagishi, and Kong Aik Lee. The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection. In *INTERSPEECH 2017, Annual Conference of the International Speech Communication Association, August 20-24, 2017, Stockholm, Sweden*, Stockholm, SWEDEN, 08 2017.

[27] Zhizheng Wu, Nicholas Evans, Tomi Kinnunen, Junichi Yamagishi, Federico Alegre, and Haizhou Li. Spoofing and countermeasures for speaker verification: A survey. *Speech Communication*, 66:130 – 153, 2015.

[28] P. V. A. Kumar, L. Balakrishna, C. Prakash, and S. V. Gangashetty. Bessel features for estimating number of speakers from multispeaker speech signals. In *2011 18th International Conference on Systems, Signals and Image Processing*, pages 1–4, June 2011.

[29] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage. Experimental security analysis of a modern automobile. In *2010 IEEE Symposium on Security and Privacy*, pages 447–462, May 2010.

[30] J. Makhoul. Linear prediction: A tutorial review. *Proceedings of the IEEE*, 63(4):561–580, April 1975.

[31] V. S. Ramaiah and R. R. Rao. Multi-speaker activity detection using zero crossing rate. In *2016 International Conference on Communication and Signal Processing (ICCSP)*, pages 0023–0026, April 2016.

[32] M. Z. Ikram. Double-talk detection in acoustic echo cancellers using zero-crossings rate. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1121–1125, April 2015.

[33] F. Gustafsson and F. Gunnarsson. Positioning using time-difference of arrival measurements. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, volume 6, pages VI–553, April 2003.

[34] K. Kwak and S. Kim. Sound source localization with the aid of excitation source information in home robot environments. *IEEE Transactions on Consumer Electronics*, 54(2):852–856, May 2008.

[35] Voice frequency. Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Voice_frequency, 2019. [accessed April 2019].

[36] Jesús Villalba and Eduardo Lleida. Detecting replay attacks from far-field recordings on speaker verification systems. In Claus Vielhauer, Jana Dittmann, Andrzej Drygajlo, Niels Christian Juul, and Michael C. Fairhurst, editors, *Biometrics and ID Management*, pages 274–285, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[37] Timbre. Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Timbre, 2019. [accessed April 2019].

[38] Maxima and minima. Wikipedia, the free encyclopedia, 2019. https://en.wikipedia.org/wiki/Maxima_and_minima, [accessed April 2019].

[39] J. M. Valin, F. Michaud, J. Rouat, and D. Letourneau. Robust sound source localization using a microphone array on a mobile robot. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 2, pages 1228–1233 vol.2, Oct 2003.

[40] F. Li and R. J. Vaccaro. Performance degradation of doa estimators due to unknown noise fields. *IEEE Transactions on Signal Processing*, 40(3):686–690, March 1992.

[41] MATLAB Function Reference. wavwrite function. http://matlab.izmiran.ru/help/techdoc/ref/wavwrite.html. Accessed September, 2019.

[42] Wen Ge, Xu Hongzhe, Zheng Weibin, Zhong Weilu, and Fu Baiyang. Multi-kernel pca based high-dimensional images feature reduction. In *2011 International Conference on Electric Information and Control Engineering*, pages 5966–5969, April 2011.

[43] Common-mode interference. Wikipedia, the free encyclopedia, 2018. https://en.wikipedia.org/wiki/Common-mode_interference, [accessed December 2018].

[44] Neousys Technology. In Vehicle Computing, 2019. https://www.neousys-tech.com/en/product/application/in-vehicle-computing, [accessed April 2019].

[45] Javier Perez Fernandez, Manuel Alcazar Vargas, Juan M. Velasco Garcia, Juan A. Cabrera Carrillo, and Juan J. Castillo Aguilar. Low-cost fpga-based electronic control unit for vehicle control systems. *Sensors*, 19(8), 2019.

[46] Pinar Muyan-Ozcelik and Vladimir Glavtchev. GPU Computing in Tomorrow's Automobiles. https://www.nvidia.com/content/nvision2008/tech_presentations/Automotive_Track/NVISION08-GPU_Computing_in_Tomorrows_Automobiles.pdf, 2019. [accessed April 2019].

[47] Electronic Control Unit. Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Electronic_control_unit, 2019. [accessed September 2019].

[48] Vehicle Audio. Wikipedia, the free encyclopedia, 2018. https://en.wikipedia.org/wiki/Vehicle_audio, [accessed December 2018].

[49] V. V. Baskar, B. Abhishek, and E. Logashanmugam. Emd-fb based denoising algorithm for under water acoustic signal. In *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, pages 106–111, July 2014.

[50] L. Wang, J. D. Reiss, and A. Cavallaro. Over-determined source separation and localization using distributed microphones. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(9):1573–1588, Sep. 2016.

[51] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. Gyrophone: Recognizing speech from gyroscope signals. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 1053–1067, San Diego, CA, 2014. USENIX Association.

[52] Youssef Bassil and Paul Semaan. ASR context-sensitive error correction based on microsoft n-gram dataset. *CoRR*, abs/1203.5262, 2012.

[53] Christophe Ris and Stephane Dupont. Assessing local noise level estimation methods: Application to noise robust asr. *Speech Communication*, 34(1):141 – 158, 2001. Noise Robust ASR.

[54] N. N. An, N. Q. Thanh, and Y. Liu. Deep cnns with self-attention for speaker identification. *IEEE Access*, 7:85327–85337, 2019.

[55] Y. Lukic, C. Vogt, O. Dürr, and T. Stadelmann. Speaker identification and clustering using convolutional neural networks. In *2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Sep. 2016.

[56] Rosa González Hautamäki, Tomi Kinnunen, Ville Hautamäki, Timo Leino, and Anne-Maria Laukkanen. I-vectors meet imitators: on vulnerability of speaker verification systems against voice mimicry. In *INTERSPEECH*, 2013.

[57] Phillip L. De Leon, Bryan Stewart, and Junichi Yamagishi. Synthetic speech discrimination using pitch pattern statistics derived from image analysis. In *INTERSPEECH*, 2012.

[58] Z. Ali, M. Imran, and M. Alsulaiman. An automatic digital audio authentication/forensics system. *IEEE Access*, 5:2994–3007, 2017.

[59] M. Zhou, Z. Qin, X. Lin, S. Hu, Q. Wang, and K. Ren. Hidden voice commands: Attacks and defenses on the vcs of autonomous driving cars. *IEEE Wireless Communications*, pages 1–6, 2019.

[60] Takeshi Sugawara, Benjamin Cyr, Sara Rampazzi, Daniel Genkin, and Kevin Fu. Light commands: Laser-based audio injection on voice-controllable systems. 2019.

[61] Tavish Vaidya, Yuankai Zhang, Micah Sherr, and Clay Shields. Cocaine noodles: Exploiting the gap between human and machine speech recognition. In *Proceedings of the 9th USENIX Conference on Offensive Technologies*, WOOT'15, pages 16–16, Berkeley, CA, USA, 2015. USENIX Association.

[62] Yangyong Zhang, Abner Mendoza, Guangliang Yang, Lei Xu, Phakpoom Chinprutthiwong, and Guofei Gu. Life after speech recognition: Fuzzing semantic misinterpretation for voice assistant applications. In *Proceedings of the 2019 The Network and Distributed System Security Symposium (NDSS '19)*. Internet Society, 2019.

[63] Deepak Kumar, Riccardo Paccagnella, Paul Murley, Eric Hennenfent, Joshua Mason, Adam Bates, and Michael Bailey. Skill squatting attacks on amazon alexa. In *Proceedings of the 27th USENIX Conference on Security Symposium*, SEC'18, pages 33–47, Berkeley, CA, USA, 2018. USENIX Association.

[64] Richard Mitev, Markus Miettinen, and Ahmad-Reza Sadeghi. Alexa lied to me: Skill-based man-in-the-middle attacks on virtual assistants. In *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, Asia CCS '19, pages 465–478, New York, NY, USA, 2019. ACM.

[65] Nan Zhang, Xianghang Mi, Xuan Feng, XiaoFeng Wang, Yuan Tian, and Feng Qian. Dangerous skills: Understanding and mitigating security risks of voice-controlled third-party functions on virtual personal assistant systems. In *IEEE SP 2019*, 2019.

[66] J. Ianniello. Time delay estimation via cross-correlation in the presence of large estimation errors. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(6):998–1003, December 1982.

[67] Yunmei Gong, Lizhi Li, and Xiaoqun Zhao. Time delays of arrival estimation for sound source location based on coherence method in correlated noise environments. In *2010 Second International Conference on Communication Systems, Networks and Applications*, volume 1, pages 375–378, June 2010.

[68] Logan Blue, Hadi Abdullah, Luis Vargas, and Patrick Traynor. 2MA: Verifying voice commands via two microphone authentication. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, ASIACCS '18, pages 89–100, New York, NY, USA, 2018. ACM.

[69] Linghan Zhang, Sheng Tan, and Jie Yang. Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, pages 57–71, New York, NY, USA, 2017. ACM.

[70] Linghan Zhang, Sheng Tan, Jie Yang, and Yingying Chen. Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 1080–1091, New York, NY, USA, 2016. ACM.

[71] U. Klein and Trinh Quoc Vo. Direction-of-arrival estimation using a microphone array with the multichannel cross-correlation method. In *2012 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pages 251–256, Dec 2012.

[72] Jie Yang, Simon Sidhom, Gayathri Chandrasekaran, Tam Vu, Hongbo Liu, Nicolae Cecan, Yingying Chen, Marco Gruteser, and Richard P. Martin. Detecting driver phone use leveraging car speakers. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, MobiCom '11, pages 97–108, New York, NY, USA, 2011. ACM.

[73] D. Ying, J. Li, Y. Feng, and Y. Yan. Direction of arrival estimation based on weighted minimum mean square error. In *2013 IEEE China Summit and International Conference on Signal and Information Processing*, pages 318–321, July 2013.

[74] H. Heli and H. R. Abutalebi. Localization of multiple simultaneous sound sources in reverberant conditions using blind source separation methods. In *2011 International Symposium on Artificial Intelligence and Signal Processing (AISP)*, pages 1–5, June 2011.

[75] L. Sun and Q. Cheng. Indoor sound source localization and number estimation using infinite gaussian mixture models. In *2014 48th Asilomar Conference on Signals, Systems and Computers*, pages 1189–1193, Nov 2014.

[76] C. Jung, R. Liu, and K. Lian. A fast searching algorithm for real-time sound source localization. In *2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pages 1413–1416, Sept 2017.