Detecting Driver Phone Calls in a Moving Vehicle
Based on Voice Features

Abstract—The use of mobile phone while driving has become a major source of distraction to drivers, leading to a large number of car accidents. In this paper, we study the problem of automatically detecting driver phone calls by monitoring smartphone activities and utilizing the vehicle on-board unit. The challenges to overcome include: i) passenger phone calls should be allowed while the calls of the driver should be blocked; ii) the detection mechanism should be phone position-independent and phone owner-independent as the driver may put the smartphone at any position in the front row and make calls via an earphone, or the driver may borrow a passenger’s phone to make a call; iii) the in-vehicle environment is noisy resulted from the operating engine, the music the driver and passenger may listen to, and the conversation between passengers and/or the driver; and iv) the computational cost at the smartphone should be light as realtime phone call detection is expected to effectively block an ongoing call to and from the driver. To overcome these challenges and achieve our objective of detecting driver phone calls, we take advantage of the uniqueness of individual’s voice feature. Through a short period of learning stage, our proposed system can recognize the driver’s voice from the collected audio data. Combined with the smartphone’s call state, our scheme can determine whether the driver is participating in the current phone call or not. Our strategy takes into account the complicated in-vehicle environment, and the proposed algorithm does not rely on the location of the phone within the vehicle nor the ownership of the smartphone, as the most existing driver phone call detection mechanisms do. We develop a client-server based system with the smartphones being the clients and the vehicle on-board unit being the server. To validate our mechanism, we perform extensive real-world experiments under different scenarios. The results demonstrate a high probability of detecting driver phone calls with a small false alarm rate.

Index Terms—Driver phone use; driver phone call detection; speaker recognition; driver distraction.

I. INTRODUCTION

The use of cell phone while driving has been identified as one of the major causes of large number of car accidents in recent years. According to a study carried out by the National Highway Traffic Safety Administration, cell phone distraction resulted in 995 fatalities and 24,000 injuries in car crashes in 2009 [1]. Governments have legislated laws that prohibit driver’s use of mobile phones while driving. However, law enforcement solely is not strong enough to keep people from using their phones when they are operating a vehicle. In this paper, we propose an effective mechanism that runs in the background of a smartphone and automatically tackles the problem.

Having a phone call is the major concern in our paper as it is one of the most common reasons for attention distraction, no matter it is a handheld or a handfree phone operation. To prevent phone conversation while driving, we can simply block the phone calls or route the incoming calls to voice mail. However, determining whether the driver is answering/making a phone call is of a great challenge since the in-car environment can be complicated, especially when there are passengers talking in the same car. Prior research has developed a number of techniques to determine whether a cell phone use is in a moving vehicle, such as cell phone handoffs [2], cell phone signal strength analysis [3], and speed measurement by Global Positioning System (GPS). These approaches typically employ apps to block the incoming and outgoing calls and texts after a cell phone is detected in a moving vehicle. Such a blocking policy can also be activated in a cell phone based on the car’s speedometer reading. These applications work only when the driver is the sole person in a car, since otherwise the passengers’ cell phones will also be blocked, which is not necessary. In practice, a large number of automobile trips include passengers, which implies that only relying on the vehicle speed detection would cause an unacceptable false positive rate in passengers’ cell phones.

Existing work differentiates between a driver and a passenger by determining whether the phone is on the driver seat area or the passenger seat area with different algorithms. An interesting approach was advocated recently by Yang et al. [4], which introduces an acoustic relative-ranging system that classifies on which car seat a phone is being used. In other words, this work transforms the driver phone call problem to the classification of whether the phone is in the driver or the passenger seat area. The underlying assumption of this work is that the driver’s cell phone must always be placed in the driver seat area. Obviously, this cannot cover the cases where the driver’s cell phone is placed in the passenger seat during a phone conversation. Actually, many drivers tend to make phone calls using earphones with their cell phones residing in the passenger seat area. Apparently, this situation is not considered by Yang et al. [4]. In this paper, we propose a mechanism that places no restriction on the phone location to improve the effectiveness of driver phone call detection in more general cases.

Our design objective is to prevent the driver phone call in a mobile vehicle, no matter where the cell phone is located or whose phone it is. There exist mainly two steps in our mechanism: 1) extracting the voice feature of the driver; and 2) conducting speaker identification. To be more specific, we compare the driver’s voice with the voice collected by the operating smartphone whenever it is on a phone call mode. By making use of the driver’s voice feature, we make sure
that the passengers’ phone calls are not influenced, while the
driver’s phone calls are blocked or re-routed to a voice mail.

To extract the voice features of the driver, we install a unidirectional microphone in front of the driver seat, for example, in the center of the steering wheel. Such a microphone has two important features: i) Super Unidirectional Pickup sharply focuses on the sound directly in front of the microphone and eliminates extraneous sounds; and ii) Phase-Tube Noise Reduction System further reduces the pickup of the sounds from the side and the rear of the microphone to help avoid noise interference. With such a microphone, we can collect the driver’s voice whenever he talks, which provides the basis for driver’s voice feature extraction. Then a speaker model is trained using the feature vectors of the driver.

After this stage, we conduct speaker identification. All the voices collected by the unidirectional microphone and the smartphones are transmitted to a computing unit (i.e., the On-Board Unit (OBU)) installed in the car head unit (board) over Bluetooth connections. The OBU is like a small computer that can do all the computation tasks, such as voice feature extraction and speaker identification. OBUs will be widely installed in future cars, since they can make a lot of complicated applications practical. Speaker recognition is an extensively studied subject in audio systems. In this paper, we adopt the Mel-Frequency Cepstral Coefficients (MFCC) [5] for feature extraction, which was introduced in early 1980s. Although various alternative features have been studied, MFCC seems to be difficult to be beat in practice. For speaker modeling, we apply the Gaussian Mixture Model (GMM) [6], [7], which has become the de facto reference method in speaker recognition.

Different scenarios have been considered to validate our design, especially when there exist other passengers talking in the car when the driver is making a phone call. We conduct extensive experiments in complicated in-car environments, which could include a driver and passengers at different seats, and background with or without music noise (engine on). The results show that our algorithm is effective and robust, and is applicable wherever the phone used by the driver is located and whose phone it is.

The rest of the paper is organized as follows: In Section II, we place our work in a broad context of driving safety, and provide an overview of the related work. Section III presents our design challenges, the overview of our design, and the detection procedure. We provide the details of the proposed driver phone call detection scheme in Section IV. Extensive experiments are conducted to evaluate the performance of our design and the results are reported in Section V. We discuss the limitations, possible extensions, and the necessity of employing OBU in Section VI. Finally, we summarize our work and conclude the paper in Section VII.

II. RELATED WORK

According to the National Highway Traffic Safety Administration, driver distractions are associated with a large number of automobile accidents. Therefore, it is strongly motivated to enable a smart human-machine interface, preventing driver assistance systems and electronics (e.g., navigators, cell phones, etc.) from stealing driver’s attention. Extensive effort has been devoted to developing driver distraction detection systems. Bergasa et al. [8] presented a nonintrusive prototype computer vision system for monitoring a driver’s vigilance in real time. In [9], an interesting approach was described to determine what effect in-car interfaces could have on driver behavior and performance. A monitoring model was introduced in [10] to measure the distraction of a driver. This model is able to detect the driver’s visual and cognitive workload by fusing stereo vision and lane tracking data. Wang et al. [11] proposed a scheme to detect driver’s phoning activity by installing a video camera on the front windshield. A similar research work published by Artan et al. [12] also takes advantage of computer vision technology except that the proposed system doesn’t require extra infrastructure, and it can be implemented on the HOV/HOT highway near infra-red (NIR) camera system.

With the development of smartphones, active research in detecting dangerous driving behaviors leveraging diverse sensors embedded in smartphones has been carried out [13], [14], [15], [16], [17]. These designs facilitate a broad area of applications. For example, the vehicle dynamics can be detected through the cell phone signal strength analysis and the embedded sensors such as accelerometers and gyroscopes [3], [18], [19], [20]. Other studies use smartphone sensors and cameras to monitor the road conditions [21], [22], [23], [24]. A smart phone system that is able to detect the driver’s texting behavior using only the inertial motion sensors and magnetometers in the smart phones was proposed in [17].

Cell phone use is a major factor for driver distraction. An increasing number of automobile accidents involving cell phone uses [1] leads to an increasing public attention to cell phone distractions [25], [26]. Apparently, phone interruptions produce negative influences on driver’s attention. Extensive research has been dedicated to mitigate driver phone distractions. Wiberg et al. [27] carried out a naturalistic study on how interruptions are handled, and developed a technology named Negotiator to reduce the attention overhead. There are active effort in the area of smart systems which reduce drivers’ burden for handling their phones, allowing a sufficient time for them to focus on the surrounding traffic. For example, in typical non-headset uses, users often need to access stored information such as personal calendar during a phone conversation. This requires them to interrupt their conversations to look at the screen. Li et al. [28] investigated a solution BlindSight to avoid the need for interruptions. Another system aiming to mitigate the problem of distracted driving was proposed in [29], in which context-awareness is utilized to implement burden-shifting, time-shifting, and activity-based sharing. All of these systems manage to reduce attention requirement from a driver, which is certainly valuable for safe driving. Our work addresses the problem from a different angle. We focus on the detection of driver phone calls, which is the most critical step in order to prevent the driver attention distraction from phone calls. Our strategy is simple: we first
determine whether or not a driver is talking over a cell phone, and if yes, block the call.

To detect driver phone call, Yang et al. [4], [30] leveraged car speakers to classify whether a phone is in a driver seat. Similarly, Wang et al. [31] introduced an approach to determine whether the phone is on the left or right side of the vehicle. All these works rely on the phone location to detect driver phone call while ignoring the fact that drivers could make phone calls with their phones placed in the passenger seats. The authors in [32] proposed to use micro-movements that differ between a driver and a passenger for driver phone call detection. However, this approach requires the smartphone to be placed in certain positions on the body. Different from the above works, we adopt speaker recognition techniques [33]–[35] to detect driver phone call. By taking advantage of the uniqueness of a driver’s audio feature, we not only improve the detection accuracy but also achieve phone-owner independency and phone-location independency.

III. SYSTEM DESIGN

To detect driver phone calls, we design an approach leveraging the driver’s voice feature as the most reliable and unique distinguishing metric, instead of relying on the phone-location or the phone-owner relationship. In this section, we introduce the design challenges, present an overview of our system, and provide a high level outline of the detection procedure.

A. Design Challenges

There are a number of challenges involved in our system design and implementation.

Robustness: The in-car environment could be very noisy. The approach for driver phone call detection should work under strong noise. The running engine, radio, and passengers all contribute to the noise level in a car. Thus, the algorithm to recognize driver phone call needs to be robust in various noisy driving environments.

Obtaining Driver’s Voice Feature: Since we use the driver’s voice feature to differentiate a driver from other passengers, we need to automatically obtain the driver’s voice feature in real time. Thus the approach for driver phone call detection should be able to determine which voice is from the driver, especially when passengers talk simultaneously in the car (over their phones or to each other).

Phone-location Independence & Phone-owner Independence: As mentioned earlier, the algorithm to detect driver phone call should be location-independent and phone-owner-independent, because a phone can be placed any where inside the car and it can belong to anyone.

Our strategy uses the driver’s voice feature to differentiate a driver from other passengers. We leverage a unidirectional microphone to capture the driver’s voice in a noisy in-car environment. Since the voice feature is independent with the cell phone location and owner, our strategy is effective no matter where the cell phone is located or whose phone it is. While other existing schemes relying on the phone location to determine driver phone call could raise a high false positive rate, our strategy is more accurate and reliable. As the driver’s voice feature is captured in real-time, we also prevent driver phone call in the circumstance when the driver borrows a passenger’s phone to make a call.

B. System Overview

Basically, our algorithm includes two sub-tasks: collecting the driver’s voice to derive his voice feature model, and comparing the voice in a phone call against the driver’s voice feature model to determine whether the speaker is the driver.

We first collect driver’s audio signals as training data, which is transformed into feature vectors by feature extraction. A speaker model is then trained using the driver’s feature vectors. In the recognition stage, we intend to identify whether the driver is the speaker of the call. To achieve this objective, we collect the unknown signals with smartphones and extract their feature vectors; then compare them against the driver’s voice feature model, which is termed as pattern matching. If a match occurs, we conclude that the driver is associated with a phone call. To be more specific, the feature vectors are compared with the driver’s voice feature model to give a similarity score. The final decision is made based on this score over a threshold $\theta$: if a similarity score is less than $\theta$, the voice is not emitted by the driver; otherwise, we consider it the driver’s voice.

C. Detection Procedure

To implement our system, we need two extra devices: a unidirectional microphone, and the On-Board Unit (OBU). The unidirectional microphone is used to capture the driver’s voice. The OBU is used for speaker identification. The detection procedure is summarized as follows:

1. Whenever a cell phone is on-calling mode, its driver phone call prevention app is automatically activated, no matter whose phone it is – we assume this app is preinstalled in all smartphones. The app sends a command to the OBU over Bluetooth to notify OBU of the phone call. Meanwhile, the on-calling phone starts recording the voice spoken into itself.

2. The OBU further sends a command to the unidirectional microphone installed in front of the driver, if the microphone is not configured to record automatically. Upon receiving this command, the unidirectional microphone starts collecting the audio signal in car. Since such a unidirectional microphone has the nice features of sharply focusing on the sound directly in front of it and eliminating extraneous sounds, it is supposed to collect the driver’s voice when the driver is talking. To validate the effectiveness of using a unidirectional microphone, we conduct extensive experiments in Section V-C.

3. After recording for 6 seconds to 8 seconds, both the cell phone and the unidirectional microphone send the collected audio signal to the OBU via Bluetooth communications.

4. The OBU derives the driver’s voice feature model based on the audio signal received from the unidirectional microphone. It also extracts the caller’s voice feature based on the audio signal received from the cell phone. Then pattern matching is applied to estimate the loglikelihood of the caller’s voice feature matching the driver’s voice feature model.
5. If the loglikelihood is larger than a threshold, we consider the caller to be the driver; otherwise, the caller is a passenger.

IV. DRIVER PHONE CALL DETECTION ALGORITHM

In this section, we detail the driver phone call detection algorithm. For each step, we discuss different possible approaches and their corresponding advantages and disadvantages. The objective is to make our algorithm suitable for for general settings by considering all possible scenarios and related issues.

To determine whether a cell phone is used by a driver is always a big challenge. As mentioned earlier, the mechanism relying on cell phone location would be unreliable, since the phone could be placed on the passenger seat areas. And the situation becomes more complicated when there are passengers talking in the same car. We believe that the voice feature is the most reliable metric to distinguish a driver from other passengers. However, this raises another challenge: how to determine which voice is from the driver? Ideally, recording a single participant's voice in a quiet environment yields the best training data for feature extraction; but the feature needs to be carried to different vehicles the driver may use. Moreover, people might be unwilling to go through such a trouble. Methods that collect training data automatically are more preferred. In this paper, we take advantage of the unidirectional microphone installed in front of the driver seat to obtain the driver’s voice.

The unidirectional microphone can be installed in the center of the steering wheel, as illustrated in Figure 1. Since a unidirectional microphone has the nice features of picking up sound well from the front while attenuating all noises in the sides and the rear, the collection of the driver’s voice can be guaranteed. Actually, the unidirectional microphone can be activated by the OBU to collect the driver’s voice whenever the driver talks. By this way we can automatically obtain a driver’s voice feature in real time from the moment the driver enters the car. If the driver does not emit any voice before a phone call is made, the on-calling phone should activate the unidirectional microphone via the OBU to collect the driver’s voice.

A. Driver Feature Extraction

After the audio signal in a car is collected by the unidirectional microphone, it is transmitted to the OBU, where feature extraction and speaker modeling are performed. For feature extraction, we employ the Mel-Frequency Cepstral Coefficients (MFCC) model. MFCCs are commonly used features in automatic speech and speaker recognition. The high level implementation steps for MFCC is outlined as follows:

1. Frame the signal into short frames.
2. Take the Fourier transform of each frame to calculate the periodogram estimate of the power spectrum.
3. Apply the mel filterbank to the power spectra, sum the energy in each filter.
4. Take the logarithm of all filterbank energies.
5. Take the Discrete Cosine Transform (DCT) of the log filterbank energies obtained above.

In the following we use an example to illustrate the above procedure step by step. We start with a speech signal, which is assumed to be sampled at 8kHz.

1) Frame the signal into 20ms frames, yielding a frame length of 0.02 * 8000 = 160 samples for a 8kHz signal. The next four steps are applied to each frame. The time domain signal is denoted by \( s(n) \). After the original sound signal is framed, we obtain \( s_i(m) \), where \( m \) ranges over 1-160, and \( t \) ranges over the number of frames.

2) Take Fourier transform to calculate the periodogram estimate of the power spectrum for each frame, denoted by \( P(t) \).

3) The Mel-spaced filterbank is a set of 20 to 40 filters (26 is a standard value). We apply 26 filters to the periodogram spectrum estimate from step 2. This gives us 26 vectors. By summing the energy in each filter, we obtain 26 numbers.

4) By taking the log of each of the 26 energy values in step 3, we are left with 26 log filterbank energies.

5) Finally, DCT is performed over the 26 log filterbank energies to produce 26 cepstral coefficients, called the mel-frequency cepstral coefficients (MFCC). These coefficients form a voice feature vector, denoted by \( x_t \).

The output of this stage is typically a sequence of feature vectors representing the test signal, \( X = \{x_1, ..., x_T\} \), where \( x_t \) is the feature vector for the \( t \)th frame, and \( T \) is the number of frames.

B. Driver Model Derivation

Gaussian Mixture Model (GMM) [6], [7] is one of the most popular models for speaker modeling. GMM is a mixture of multivariate Gaussian components. For a D-dimensional feature vector (in our case, \( D = 26 \)), the mixture density \( x \) is a weighted sum of \( K \) component densities, which can be described as follows [36]:

\[
p(x|\lambda) = \sum_{i=1}^{K} w_i p_i(x)
\]

(1)

where \( K \) is the number of Gaussian components, \( w_i \) is the mixture weight of the \( i \)th Gaussian component constrained by

\[
\sum_{i=1}^{K} w_i = 1 \quad (w_i > 0).
\]

And \( p_i(x) \) is the component density of

\[
p_i(x) = \frac{1}{(2\pi \sigma_i^2)^{D/2}} \exp\left(-\frac{1}{2\sigma_i^2} x^T \Sigma_i^{-1} x\right) 
\]

where \( \Sigma_i \) is the covariance matrix of \( i \)th Gaussian component. \( \lambda \) is the parameter matrix of the GMM model.
well as those of the passengers, are clients and the OBU is the client-server based framework. The driver’s smartphones, as

A. System Requirements

p_i(x) = \frac{1}{(2\pi)^D/2|\Sigma_i|^{1/2}}exp\{-\frac{1}{2}(x - \mu_i)^T(\Sigma_i)^{-1}(x - \mu_i)\} (2)

with a $D \times 1$ mean vector $\mu_i$, and a $D \times D$ covariance matrix $\Sigma_i$.

The complete Gaussian mixture density is collectively represented by the mixture weights, mean vectors, and covariance matrices from all component densities, as denoted by the notation

$\lambda = \{w_i, \mu_i, \Sigma_i\} \quad i = 1, ..., K \quad (3)$

For speaker identification, each speaker is represented by a GMM and is referred to by his/her model parameters $\lambda$ [6]. In this study, we employ the well-established maximum likelihood (ML) estimation [6] a diagonal covariance matrix for each Gaussian component to estimate the model parameters of the GMM $\lambda$.

C. Speaker Identification

If a cell phone is on calling mode, we let the cell phone record the audio signal taken as an unknown, and transmit it to the computing unit OBU. Next, we determine whether the signal is from the driver. Speaker identification is applied to the collected unknown audio data. Mathematical description of this procedure is illustrated as follows:

First, the voice feature $X'$ for the caller is extracted through the MFCC model. The objective is to find the posterior probability for a given observation sequence $p(X'|\lambda)$. Then by using the logarithms and the independence between observations, the speaker identification system computes

$Y = \sum_{i=1}^{T} \log p(x'_i|\lambda) \quad (4)$

in which $p(x'_i|\lambda)$ is given in Equation (1). The value of $Y$ is finally compared with a threshold $\theta$. Specifically, if the value of $Y$ is higher than the threshold, a match exists, and the caller is the driver, which is described as follows:

$\begin{cases} 
H_0, & Y < \theta; \\
H_1, & Y \geq \theta 
\end{cases} \quad (5)$

where $H_0$ and $H_1$ denote the hypothesis that the speaker is not the driver and that the speaker is the driver, respectively.

Apparently, the larger the probability $p(X'|\lambda)$, the larger the value of the loglikelihood $Y$. In other words, when the speaker is more likely the driver, the value of the loglikelihood is larger. Motivated by this fact, We take the loglikelihood $Y$ as the similarity score between the driver model and the tested speech. Since $p(X'|\lambda) \in [0, 1]$, we have $Y \in (-\infty, 0]$.

V. SYSTEM IMPLEMENTATION AND EVALUATION

A. System Requirements

Our driver phone call detection system is developed on a client-server based framework. The driver’s smartphones, as well as those of the passengers, are clients and the OBU is the server. The clients we use in our experiments are Galaxy S4 smartphones that is running Android Operating System version 4.3. This cell phone model supports Bluetooth such that the smartphone is able to communicate with the server. The server is the OBU that is embedded in future automobiles. The car we use to conduct experiments is a 4-door mid-sized 2012 Mazda 3. We use an Ideapad Y400 laptop to substitute the OBU as our vehicle model is not equipped with an OBU. In addition, as mentioned in previous sections, a unidirectional microphone is installed right in front of the driver’s seat on the dashboard so that it can capture the driver’s voice and attenuate sound that comes from other directions.

B. System Implementation

As mentioned before, the system is composed of a server and one or more clients. The software on both the clients and the server are developed in Java with JAVA SDK 7 or later, while the server is built on Struts 2 web server framework and the clients run with Android API 18 or up. The unidirectional microphone is mounted in front of the driver on the dashboard waiting for recording signals from the server. For better elaboration, we call the audio signals collected by the unidirectional microphone driver’s voice, denoted by $VoD$, and those collected by the clients is denoted by $VoC$.

The server is responsible for listening to HTTP requests, collecting audio signals from both the unidirectional microphone and the clients, and performing voice feature matching. HTTP requests carry the information about clients asking the server to command the unidirectional microphone to start recording the driver’s voice. After recording the audio for a period of time, both the clients and the unidirectional microphone send their recorded audio data to the server. Here, we outsource the computational cost to the server in order to reduce the power consumption and memory cost on the clients and enhance the real-time experience. The server then starts performing MFCC on the $VoD$ to extract the driver’s voice feature. At the same time, GMM is operated on $VoC$, and the result is used to compute against the obtained $VoD$ feature vector to get the loglikelihood, which we use to determine the similarity between $VoC$ and $VoD$.

A client keeps monitoring its call states and starts to record the input sound signal, i.e. $VoC$, when a phone call is activated. As soon as the phone call state switches to "activate", the client sends a signal to the server via HTTP request asking the server to activate the unidirectional microphone to start recording the driver’s voice. The client then sends its $VoC$ to the server. As soon as the phone call gets cut off, the recording process stops.

With regards to the parameter settings, we set the speaker recognition parameters according to [6]. The audio sampling rate is 8kHz; each audio signal is framed into 20ms short-time segments; the filterbank is a set of 26 filters; the number of Gaussian components $K$ is 12; the initial weight $w_i$ for the $i$th Gaussian component is $1/12(1/K)$. As for the recording time, we conduct experiments with time interval set to be 2s, 4s, 6s, 8s, 10s, 12s, 14s, and 16 on 30 pieces of audio inputs.
As shown in Fig. 2, the fastest GMM construction time is achieved when the audio input lasts at least 8s. Therefore, we set the audio recording time to 8 seconds. The experiments are averaged over 30 runs.

![Fig. 2. Average recording time among 30 different audio inputs.](image)

C. System Evaluation

We label the driver seat, front-right seat, rear-left seat, and real-right seat as position $P1$, $P2$, $P3$, and $P4$, as illustrated in Fig. 1. To demonstrate the robustness and generality of our strategy, we conduct experiments with various scenarios and settings as listed below.

- There is no passenger in the car.
- There is one passenger who sits in $P2$, $P3$, or $P4$.
- There are two passengers. One passenger sits in the front seat $P2$, and the other one sits in $P3$ or $P4$.
- There are three passengers sitting in the car.

We also conduct experiments with music or radio as background noise. The volume is set to be 40dB, 50dB, and 60dB. We evaluate the system from the following three perspectives:

- The effectiveness of the unidirectional microphones;
- Driver phone call detection delay;
- Driver phone call detection accuracy.

All the results are obtained when the engine is on, mimicking a driving environment. The driver is talking over the phone with someone while the passenger(s) are either reading or taking a phone call, too. The plotted curves in the following subsections represent the averaged results over 30 runs.

**Effectiveness of the unidirectional microphones.** We choose three different unidirectional microphone models. They are listed as follows,

- Mic#1-Sony Unidirectional Camcorder Microphone
- Mic#2-Sound Professionals USB High Sensitivity unidirectional Lapel Microphone, and
- Mic#3-Andrea Electronics Shotgun Microphone (SG-100).

We test each microphone model under the aforementioned settings and compare the average performances. We also compare the performance with that of an omnidirectional microphone, Sony ECMCS3 Clip style Omnidirectional Stereo Microphone. To test the performance and necessity of unidirectional microphones, we obtain the driver’s voice with and without passenger talking because we want to focus on how much the unidirectional microphone can filter out the voice outside it’s receiving range. We use the similarity score between $VoD$ and $VoC$ defined in Section IV-C to quantify the performance. Note that the ideally obtained driver’s voice is only used here to test and compare the performance between unidirectional microphones. For our system evaluations, the driver’s voice is recorded in realtime in a more complicated in-car driving environment. For each test, we mount one of the Mic#1, #2, #3 and the omnidirectional microphone in front of the driver as shown in Fig. 1.

![Fig. 3. (a) Average performance of unidirectional microphones without and with a passenger reading at different seat; (b) performance comparison of the microphones without and with a passenger reading at different seat; and (c) the impact of background noise on the similarity score.](image)

Fig. 3(a) depicts the similarity score between the driver’s voice $VoD$ and the $VoC$ that consists of the driver’s voice and one single passenger’s voice (reading a book). The higher similarity score means the driver’s voice is less impacted by the passenger. Two observations draw our attention.

First, one can see from the figure that the similarity score is much higher when the driver is the only talker than that when passengers are present. Second, the plot also shows that position $P2$ has the lowest similarity score, which means that the driver’s voice is affected by the front-right seat the most, while the voices emitted from $P3$ and $P4$ do not affect the similarity as much as the front passenger’s voice. This is because the voice from position $P3$ is blocked by the driver’s seat and $P4$ is the farthest to the microphone. The sound from $P2$, on the contrary, has no problem traveling to the microphone and it has the shortest distance to the microphone compared to $P3$ and $P4$.

Fig. 3(b) gives a sharp comparison between unidirectional and omnidirectional microphones when there is one passenger in the car sitting in different seats reading a book. The plot indicates that omnidirectional microphone performs the best given that the driver is the only talker in the car, but it gives much poorer result when there is passenger talking. Still, the omnidirectional microphone doesn’t perform as well as the unidirectional microphones since the omnidirectional microphones we use have the functionality to automatically enhance the voice that is in their listening range while restraining voice from other directions. Fig. 3(b) also depicts the same impacts of the passenger’s voice as Fig. 3(a) that $P2$ affects the driver’s voice the most while $P3$ and $P4$ have similar influences.

Fig. 3(a) and Fig. 3(b) clearly demonstrate that unidirectional microphone is more effective and critical to the configuration of the driver phone call detection system. Note...
that, microphone #2 gives the highest similarity score in all the above experiments.

To further study the impact of background noise on the unidirectional microphones, we carry out experiments with music playing in the background at different volumes when the driver is the only talker in the car. We conduct three sets of experiments with background music volume tuned at 40dB, 50dB, and 60dB. Fig. 3(c) shows that the similarity score decreases by 1 as the music volume increases by 10dB. When the volume of the background music reaches 60 dB, it is loud enough to cover the sound of people talking, however, the similarity score is still above $-11$. In this case, people will need to turn down the music or raise their voices. Therefore, the loud background noise does affected the similarity score, but the extent of the influence is limited.

Our experimental results do show that i) unidirectional microphones are indeed more effective than omnidirectional microphones; ii) voice from passenger who’s sitting $P2$ has the most impact on the driver’s noise; and iii) unidirectional microphones can still successfully pick up the driver’s voice even with noise in the background. Since microphone #2 gives the best performance among all the three unidirectional microphones, we will continue evaluating our system with microphone #2.

**Driver phone call detection delay.** As mentioned in Sec V-B, the system is built-on a client/server framework. The software running on the clients is an Android application pre-installed on the smart phones. The clients start to record the audio input as soon as a phone call is activated. After 8 seconds of recording, clients send the audio data to the server, the OBU. OBU starts the computation on both audio data received from clients and unidirectional microphone to get the similarity score. OBU then sends back the results to the clients, and the clients will take actions accordingly.

In our design, audio processing is out-sourced to OBU, which leads to a reduction on the computational overhead at the client side. Meanwhile, data transmissions between clients and OBU, and between OBU and the unidirectional microphone introduce extra time for the clients to wait before they can cut off the current phone call. Therefore, there is a detection delay in our system. The delay mainly consists of three parts, the duration of the audio data uploading time, processing time, and feedback time. We consider the worst case scenario where all the passengers are making phone calls at the same time. We run 60 times of an experiment consisting of a driver and 3 passengers making phone calls at the same time. For each run, we record the duration of the three parts and the summation of them. Fig. 4 plot the time durations and the total time delay for the 60 runs where one can see that the total delay time is within $2200ms$ and $2600ms$, which means that it takes no longer than $3s$ for the system to detect a driver phone call and send out alert messages or block the call.

**Driver phone call detection accuracy.** Based on our design the detection system will cut off the phone call if the similarity score is higher than a certain threshold. This requires a threshold that gives the best tradeoff between the true positive rate and the false positive rate of the system. Hence, we resort to Receiver Operating Characteristic (ROC) curve to select the best operating value as the threshold.

ROC curve is the result of plotting the true positive rate ($TPR$) against the false positive rate ($FPR$) over a wide range of thresholds [37], with $TPR$ being the ratio of cutting off the phone call when the driver is actually on the call, and $FPR$ being the ratio of cutting off the phone call when the driver is not at all involved. We conduct experiments under the scenarios listed before in Sec V-C where there is 0, 1, 2, 3 passengers talking or making phone calls while the driver is taking a phone call, and scenarios where there is background music at 40dB, 50dB and 60dB. For each scenario and each threshold point, we calculate the $TPR$ and $FPR$ and plot the result in Fig. 5.

In the ROC curve shown in Fig. 5, the most upper left coordinate $(0,1)$ represents 0% false positive rate and 100% true positive rate [38]. Theoretically, we should choose a threshold that corresponds to the point that’s closest to point $(0,1)$. From the plot, we get two thresholds that yields equally good $TPR$ and $FPR$ ratio, they are $-10.7$ with detection accuracy of $85\%$ and threshold $-10.6$ with detection accuracy of $86\%$. We choose the threshold $-10.6$ for the higher accuracy.

With the threshold being finally settled, we further evaluate our system performance by the true positive rate and the false positive rate from three aspects: the impact of the passenger position, the impact of the background noise, and the impact of multiple passengers. The following experiments are conducted with the best threshold value $-10.6$. 

![Fig. 4. System delay in detecting driver phone call.](image1)

![Fig. 5. ROC curve under difference circumstances.](image2)
The impact of different passenger position
As shown in Fig. 6(a) and Fig. 6(d), the front-right seat affects the system the most. When the passenger sits and talks in the front-right seat the similarity is the lowest and FPR is the highest, which makes sense since the voice from this position can directly reach the unidirectional microphone and the audio signal strength is pretty high since there is no blocking on the way. Between the rear seats, the passenger sitting in the rear-right seat shows the smallest FPR rate though the similarity score under this scenario is not as high as that when the passenger is sitting in the rear-left seat. This is mainly because the sound from rear-left seat gets blocked by the driver’s seat, which means the audio signal is weaker when it reaches the unidirectional microphone than the signal that is emitted from the rear-right seat. Nonetheless, the system’s true positive rate (TPR) is still above 98% for all the three cases.

The impact of background noise
Fig. 6(b) and 6(e) report the impact of background noise measured under three scenarios where we set the background music at 40dB, 50dB, and 60dB. Again, we use the threshold chosen based on the ROC curve. From these figures one can see that the louder the background music, the more affected the system is. The similarity score drops below −10.6 when the background noise is 60dB. FPR and TPR suffer the same extent of impact by the 60dB noise. However, we do not consider this phenomenon a big issue as loud background noise will force the speakers to raise their voices. Even though the system is affected a lot by the loud background noise, our system still obtains over 90% true positive rate.

The impact of multiple people talking
Multiple people talking could intuitively affect the system performance. We conduct experiments when three people or four people, including the driver, talking at the same time and the results are plotted in Fig. 6(c) and Fig. 6(f). We observe that the more people who are talking, the more affected the system is. One can see that the similarity score shows similar result to that of the background noise. The TPR value of three people talking is 80% and 67% for 4 people talking. Although the TPR is not satisfying, the FPR is effectively restricted to 10% and 15%, and this situation of multiple people talking at the same time rarely occurs in small vehicles in real life.

VI. DISCUSSION

Limitation. Even if a driver phone call can be detected within 8 seconds, it still brings up some distraction to the driver when the driver picks up the phone. However, we believe our approach can significantly improve the driving safety by timely eliminating the phone call distraction compared to other approaches. Another limitation is that we leverage some extra hardware such as an inexpensive unidirectional microphone and an OBU to achieve driver phone call detection. This should not be a concern as future smart vehicles have OBU installed and vehicle manufactures can easily and be willing to equip a vehicle with a unidirectional microphone in its steering wheel if this helps to enhance safety.

Extension. Our system can be integrated with other driver assistant systems and additional vehicle sensors to make it a more complete one. For example, before determining whether a phone call is made by the driver or a passenger, speed detection technique should be applied to determine whether the cell phone is in a moving vehicle. Once a driver phone call is detected, other techniques to reduce drivers’ effort for handling their phones to avoid the need for interruption can be triggered. A warning for vehicle control could be raised too. Such systems are generalized as Advanced Driver Assistance Systems (ADAS) [39].

VII. CONCLUSION

In this paper, we propose an effective mechanism to prevent driver phone call by taking advantage of the uniqueness of the driver’s voice feature. Different from the existing works that either depend on phone location or confuse driver phone call with passenger phone use, our algorithm achieves a high detection rate and a low false-alarm rate at different complicated scenarios. We can detect driver phone call wherever the phone is placed and whoever the phone belongs to when there are passengers talking simultaneously. To further demonstrate the robustness and generality, we also consider the impacts of different factors, such as position and noise in our experiments. The results show that our strategy guarantees the driver phone call prevention with a high probability and reliability.

In our further work, we intend to integrate our driver phone call detection mechanism with other driver assistance systems to further enforce the user driving safety behavior. We also will consider driver texting/messaging detection via image processing techniques.

REFERENCES
