In-Vehicle Phone Localization for Prevention of Distracted Driving

Chun-Yu Chen and Kang G. Shin, *Life Fellow, IEEE*

Abstract—A new in-vehicle phone localization scheme, called DAPL (Detection and Alarming of in-vehicle Phone Location), is proposed to determine the locations of smartphones inside a moving car, with the goal of preventing smartphone-distracted driving. DAPL operates on commodity smartphones and cars, and does not require additional special/dedicated hardware to be installed inside the car, making its deployment easy and attractive to users and carmakers. Even when a phone is moved from one location to another inside a moving car, DAPL will detect this movement, acquire the sensor data, and estimate the phone's destination location. DAPL captures the trajectory of each phone movement, the change of magnetic field, and the RSSI readings from the Bluetooth transceiver built in most cars, and then estimates the phone's destination location by matching the trajectory with the variation of magnetic field and the Bluetooth RSSI readings. Our extensive experimentation has shown DAPL to achieve an average of 91.71% accuracy of in-car phone localization at low energy overhead, i.e., $<2.5\%$ reduction of actual usage (screen-on) time.

Index Terms—Applications, mobile computing, distracted driving, mobile sensing, in-car phone localization

1 Introduction

While embracing the convenience brought by mobile devices/apps, people use their mobile devices wherever they go and whenever they want. *Phone-distracted driving* is an act of using a smartphone while driving a car, such as texting or browsing web. Recent statistics show that the likelihood of getting involved in car accidents increases 23x when drivers are texting [1], making texting one of the top causes of car accidents [2], [3]. To make things worse, texting while driving is known to be a common behavior of teenage drivers [4].

Considering its grave danger, how to prevent/reduce phone-distracted driving has become critically important for safety. Various solutions have been proposed to mitigate phone-distracted driving, which can be classified as *Detection* or *Prevention* approaches (Section 2). The former [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17] is designed to detect the occurrence of phone-distracted driving by monitoring the driver's behavior to detect any feature/sign of distracted driving. If yes, the system will warn the driver or disable distracted functions (e.g., texting, web browsing, notifications, etc.). It detects the distracted driving caused not only by smartphones but also by general activities, such as eating. However, it cannot prevent phone-distracted driving because its detection requires the behavior of distracted driving to occur at least once.

On the other hand, prevention approaches [18], [19], [20], [21], [22], [23], [24], [25], [26] aim at disabling the distractive use of phones if they are in the driver seat. According to their functionalities, these approaches can be classified further as *Initial Location Identification* (ILI) or *Location Tracking* (LT). ILI [18], [19], [20] focuses on monitoring the sensor data when the phone enters the car, and checks if any feature associated with a certain seat is detected in order to determine the phone location. It has the advantage of not requiring any additional dedicated hardware/device to operate. However, it cannot track the phone location inside the car because its detection is based on features that usually occur only once per trip (e.g., the event of entering car).

Location tracking of phones inside a car is crucial for two purposes. The first purpose is to *automatically* mitigate or prevent phone-distracted driving because the drivers often retrieve phones from their pocket, purse or bag and then use them while driving. When entering their car, a large percentage of people first put their purses/bags (with phones inside[1]) in the passenger seat, retrieve phones from their purses/bags upon hearing a notification sound, and then read & reply to text messages, all while driving. Also, drivers often place their phones on a phone stand or cup holder after entering the car. Upon hearing a notification sound, the driver usually picks up (or looks at) the phone in the phone stand or cup holder, and then reads/replies to text messages. The inability of phone location tracking inside a car will leave these common/frequent use-cases undetected, thus endangering safety. These are common (*not corner*) cases. Furthermore, location tracking allows the system to mitigate the effect of incorrect initial phone localization.

The second purpose is to ensure user convenience. That is, the system should allow the use of the phone if/when it is no longer in the driver seat. For example, the driver may take the phone out of his pocket or from the phone stand/cupholder and hand it to the passenger to find information

• *The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2121 USA. E-mail: {chunyuc, kgshin}@umich.edu.*

Manuscript received 31 Dec. 2020; revised 20 Dec. 2021; accepted 27 Dec. 2021. Date of publication 6 Jan. 2022; date of current version 5 May 2023.
This work was supported in part by NSF under grant CNS-1646130.
(Corresponding author: Chun-Yu Chen.)
Digital Object Identifier no. 10.1109/TMC.2022.3141019

1. More than 80% of female users aged 15–40 in Melbourne [27] and $>60\%$ female users in 11 cities world-wide [28] are reported to keep phones in their purses/bags when they are not using them.

or reply to messages on his behalf. If the phone does not automatically enter/exit the driving mode, the driver may have to manually perform these actions (or instruct an accompanying passenger how to do them), causing another potential distraction.

Techniques similar to in-door localization are utilized for LT [21], [22], [23], [24], [25], [26] to directly locate the phone inside a car. Since the phone is in a moving car, they usually require additional *dedicated hardware*, which acts as anchor(s) or reference point(s). Since such hardware is originally not included in contemporary cars, owners/users need to install it in their cars. Also, cost-conscious car manufacturers are not likely to provide the additional hardware for free [29], [30], thus incurring cost to users [31] and making its deployment less attractive because those who use phones while driving are unlikely to purchase a product that will limit their phone usage. To the best of our knowledge, the only LT approach [26] that does not require additional hardware can only determine whether the phone is in the left/right side of the car, but cannot distinguish between front and back row seats without upgrading the in-car audio system. Both deployability and full functionality are the keys to wide adoption of distracted driving prevention solutions. However, the solutions proposed thus far can only achieve limited functionality without relying on dedicated hardware or modification to the in-car hardware.

Due to the lack of *automatic* mechanisms that are both easy to deploy and able to locate the phone in real time, modern phone OSes or apps [32], [33] still depend greatly on users to *manually* enable/disable the driver mode or to connect the phone to in-car systems . To bridge the gap between deployability and full functionality, we propose a novel LT scheme, called DAPL (Detection and Alarming of in-vehicle Phone Location). Unlike prior LT schemes, DAPL eliminates the need for additional dedicated anchors/hardware for providing spatial references to phone location. Instead, DAPL utilizes the temporal features of sensor data induced by phone movements to determine its destination location. DAPL is designed to operate on smartphones and the only requirement of its deployment is to enable the beacon mode of the built-in Bluetooth transceiver in a modern infotainment system.[2] However, we will show DAPL is still capable of performing localization without in-car Bluetooth in Section 4. By utilizing the commodity hardware of smartphones and cars, DAPL can thus be deployed with a simple software installation. Comparison of DAPL with related approaches is summarized in Table 1 and discussed in Sections 2 and 4.

The goal of DAPL is to enable phone and app developers to determine whether the phone is in the driver seat or in the passenger seats. DAPL exploits the fact that different phone movements produce different features embedded in the data gathered from the phone's inertial measurement units (IMUs). It utilizes sensor fusion to extract features from the collected sensor data and estimates the phone location by matching the features with a specific movement. To ensure good user experience, we also develop a mechanism to reduce DAPL's energy consumption while achieving good accuracy.

TABLE 1
Comparison of Related Works, Where (✓) Means the Function is Only Valid When the Phone Stays Still at its Original Location

	Detection		Prevention				DAPL
	[5]–[15]	[16], [17]	[18]–[20]	[21]	[22]–[25]	[26]	
First Time Prevention	×	×	(✓)	(✓)	✓	✓	✓
Location Tracking	N/A	N/A	×	(✓)	✓	✓	✓
Multiple Phone Tracking	N/A	N/A	×	✓	✓	×	✓
No need of wearable or extra dedicated hardware	×	✓	✓	✓	×	✓	✓
No need of hardware modification to cars	✓	✓	✓	✓	✓	×	✓
No need of controlling subsystems in cars	✓	✓	✓	✓	✓	×	✓

For prevention/reduction (*not* detection) of phone-distracted driving, DAPL provides two functionalities. The first (major) functionality is that DAPL supports phone OSes to *actively* disable certain functions (e.g., notification and texting) while the phone is determined to be in the driver seat. Since a notification sent to drivers can also cause them to take their eyes off from the road for as long as 5s (e.g., reading a text [35]), DAPL's automatic activation can also reduce the "unintended" distracted driving due to text notifications even if the driver is aware of the danger of distracted driving and does not use the phone voluntarily. The secondary functionality is that DAPL can potentially be a tool to *passively* help drivers gradually quit the habit of using phones during driving as a long-term safety effect. Because DAPL creates a barrier for drivers to use their phones, the drivers who are aware of the danger of phone-distracted driving but cannot help it may potentially abandon the use of their phones and eventually not use them during driving. However, DAPL is *not* intended for adversarial scenarios where users are "determined" to use their phones and evade DAPL's detection because there is no incentive for them to install DAPL or use phones with DAPL in the first place.

Even though DAPL is motivated by prevention of phone-distracted driving, it is essentially designed to determine whether or not the phone is in the driver seat. The novelty of DAPL lies in enabling the functionality of location tracking in a mobile/noisy environment based only on the existing features in the sensor data without requiring any hardware modification or additional dedicated device (i.e., the barriers that reduce the incentive of using the system). We must meet the following three technical challenges to achieve this.

C1. Localization in a Mobile Environment. Without the aid from dedicated anchors, the phone must determine 1) whether it has moved inside the car or just moved along with the car, and 2) where it moved to, all in a mobile environment (i.e., a non-inertial frame system) where the sensor readings are affected greatly by the motion of the car.

C2. Enhancement of Noise Resilience. There can be excessive noise generated by the operation of cars and/or the surrounding environment [36], [37]. Relying solely on one or two sensor measurements can lead to a large number of false positives/negatives.

C3. Reduction of Energy Consumption. Since DAPL is intended to run on a single, standalone phone without requiring any direct network connection, all information gathering

2. 86% of vehicles shipped in 2018 come with built-in BT [34].

and calculation are limited to the phone itself. The phone must be tracked without consuming too much energy to preserve good user experience.

This paper makes the following contributions in meeting the above challenges:

- Development of phone location tracking, DAPL, without requiring any additional dedicated device and direct communication with subsystems in a car (Section 3):
  - (Meeting C1) Development of in-car movement detection (Section 3.3), and movement trajectory estimation in a mobile environment (Section 3.4).
  - (Meeting C2) Development of a noise-resilient scheme for in-car phone localization (Section 3.5).
  - (Meeting C3) Proposal of an energy-saving mechanism for practical use scenarios (Section 3.6).

The rest of the paper is organized as follows. Section 2 discusses the prior work related to DAPL. The detailed system design is then presented in Section 3. We demonstrate DAPL's accuracy via extensive experiments (500+ test cases) in real driving scenarios with 91.71% localization accuracy, 3.80% false positive, and 4.49% false negative rate in Section 4. Section 5 describes the evaluation results for the energy consumption of DAPL. Finally, the paper concludes in Section 6.

2 Related Work

Existing applications (e.g., AT&T DriveMode [38] and Sprint Drive First [39]) and built-in driving mode in modern phone OSes [32], [33] can only determine whether the phone is in a car (based on the moving speed [32], [33], [38], [39] or connection to in-car infotainment systems [32], [33]) but cannot identify whether the phone is used by the driver. Little has been done to actually detect/prevent distracted driving, and hence researchers still seek solutions to this problem.

2.1 Detection Approaches

These approaches are intended to detect the occurrences of distracted driving. They usually require additional devices to monitor drivers' motion/posture to determine whether or not the driver is attentive to driving. Specifically, [5], [6], [7], [8] utilize smartwatches to capture the hand motion of drivers to determine whether they are attentive to driving based on the steering movements. Deshmukh *et al.* [9] proposed use of electrocardiogram to detect certain features that indicate distracted driving. Machine learning is used to analyze drivers' posture (e.g., head movement) based on the video recorded by in-car cameras [10], [11], [12], [13], [14] or Wi-Fi signals [15].

There have been other proposals that do not require additional hardware, to function. Watkins *et al.*[16] proposed use of smartphone usage patterns (e.g., typing speed and gaps between entering each character) to determine whether the user is driving or not. Driving features [17] (e.g., how the user steers his/her vehicle) are extracted from the in-car signals to determine if the driving behavior deviates from the normal/attentive driving pattern.

The detection approaches have the advantage of covering general distracted driving scenarios, but they require a certain distracted driving event to happen for detection, thus limiting their functionality. As a result, they are good at raising the awareness of distraction, but not designed to prevent phone-distracted driving.

2.2 Prevention Approaches

2.2.1 Initial Location Identification (ILI)

ILI approaches [18], [19], [20] determine a smartphone's location by determining (i) which side of the car the user enters and (ii) the type of phone user by identifying driver/passenger-specific features. TEXIVE [18] uses the acceleration and angular speed when the user enters the car, to determine the direction (i.e., the left side or the right side) of entering the car. After determining the user's entry side, it monitors the acceleration when the car goes through bumps to distinguish whether the phone is located in the front or the back row. Similarly to TEXIVE, Chu *et al.* [19] proposed an approach to determine the phone position utilizing the motions of entering the car. Along with the detection of the pedal pressing and buckling the seat-belt motion, it can determine whether the phone is in the driver seat or not. The user's motion of sitting down and the sound of closing a door are utilized by Park *et al.* [20] to detect the event of entering the car and determine the entering direction. Magnetic-field fluctuations caused by starting the engine are then used to determine whether the phone is in the front or the back seat.

ILI approaches do not require any additional device to be installed inside cars, but their detection of phone location is limited to the phones that enter the car with the user. They cannot track the phone location if the phone moved from its initial location after entering the car, e.g., the driver retrieves his phone from the front passenger seat.

2.2.2 Location Tracking (LT)

These approaches [22], [23], [24], [25], [26] determine a phone's location by methodologies similar to indoor localization. Bluetooth beacons are installed in each car seat to serve as localization anchors [22]. Phones then estimate their own location by identifying the largest RSSI readings from those anchors. Wang *et al.* [23] proposed to determine the phone location by comparing its centripetal acceleration and a reference device installed in the middle of the car when the car makes a turn. This approach only distinguishes whether the phone is in the left/right side of the car without determining whether the phone is actually in the front row seat. Wahlstrom *et al.* [24] proposed to determine the phone location by fusing data collected from a smartphone (with both gyroscope and accelerometer) and at least 1 additional accelerometer acting as the anchor with known location. Specifically, the system estimates the relative positions between the smartphone and the anchor accelerometer(s) by formulating the estimation problem as a nonlinear filtering problem. Wang *et al.* [25] proposed the installation of a resistive seat cushion in the driver seat so that the mobile device will detect the special pattern induced by capacitive coupling when the driver touches the device.

Sound-based phone localization was proposed in [26], where the phone is used to control the audio system in the car and play a different sound from each speaker in the car.

Phone location is determined by measuring the time difference of arrival (TDOA) of different sounds when they were played simultaneously from different speakers of the car. However, since there are still only 2 *controllerable* audio channels instead of 4 in basic model cars, it can only be used to determine whether the phone is at the left/right side of the car. Also, [26] requires direct Bluetooth connection to the car audio subsystem, which limits the detection to only the connected phone. Theoretically, one can combine [26] with another ILI for detecting an event of entering the car and remove the connection requirement. However, since the phones without connection will not know when the sound is emitted, they have to continuously record and process the received sound, which, as the authors of [26] stated, is not their intended design owing to its substantial energy consumption.

The approaches proposed in [22], [23], [24], [25], [26] require extra dedicated hardware or premium upgrade to the audio system for full functionality and, therefore, incur additional cost and installation efforts to OEMs or vehicle owners, thus rendering their deployment less attractive [29], [30]. In contrast, DAPL exploits the detection of users entering the car and the initial phone location information obtained from [20], and tracking, in real time, the phone movements inside the car, without the requirements of prior LT approaches.

Johnson *et al.* [21] proposed to identify the phone location based on the features of roll and pitch measurements when the car decelerates or makes a turn. [21] is similar to DAPL in a way that it does not require any dedicated anchor device to be installed in the car. However, as authors of [21] stated, their detection is limited to the case when the vehicle is travelling on a flat terrain and the phone is maintaining the same orientation (e.g., staying still) within the vehicle. This limitation and its detection mechanism (i.e., relying on the occurrence of certain vehicle movements for its detection) make [21] unable to handle the common case where the driver retrieves (gets) the phone from another seat (person) and start to use it (i.e., one of the most important/common cases DAPL is designed to handle). From a functional perspective, [21] is closer to a *detect-then-block* scheme after the phone is in the driver's hand for a while, but less of a scheme to prevent drivers from using their phone in the first place.

2.3 Comparison of DAPL With Prior Work

Table 1 shows a functional comparison between DAPL and prior work. As a prevention approach, DAPL aims at preventing occurrence of smartphone-distracted driving based on location tracking, instead of stopping it after it has already occurred. It features easy deployment without the limitation of only one phone per car, nor requirements of extra dedicated hardware, nor hardware modification to the car. From the technical perspective, DAPL uses temporal features for location tracking in noisy environments, instead of conventional fingerprinting (i.e., static signatures or spectrum analysis) utilized in prior localization approaches. Specifically, DAPL exploits the temporal features associated with in-car movements for determining a potentially coarse-grained trajectory to estimate the phone's location. That way, the system will not rely on transient measurements (which are susceptible to the measurement environment) for its location tracking, making it resilient to transient noises. Furthermore, multiple features are utilized simultaneously (as redundancies) to mitigate the potential failure of feature extraction or absence of (sensor) data.

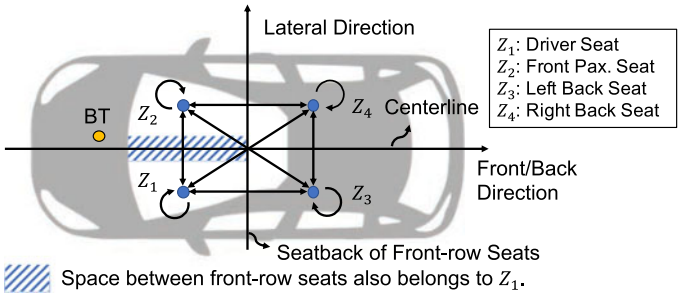


Fig. 1. Possible phone movements and in-car zones (Z_1–Z_4), where space between front-row seats also belongs to the driver seat (Z_1).

Finally, also the most important of all, the goal of DAPL is to reduce the occurrence of smartphone-distracted driving by providing the functionality of phone location tracking *with minimum deployment requirements possible*. We believe enabling wide adoption will be more effective than having a slightly better accuracy with less incentive for deployment. We will use a simple example to illustrate this concept. Assuming a person may use his/her phone while driving 10 times per day, a system with 90% accuracy can prevent 9 times from happening per day. If one more person will be willing to try DAPL, it can prevent 9 more occurrences from happening even if we assume DAPL makes no performance improvement.

3 System Design

3.1 System Overview

Presented below is the fundamental concept of DAPL and its system architecture. DAPL utilizes the fact that different phone movements between seats will have different trends/features embedded in the collected sensor data (instead of directly using the raw data). DAPL determines the phone movement between seats (Fig. 1) based on the extracted features to estimate the phone's location. Note that DAPL considers multiple features simultaneously to neutralize the condition under which some of them may be affected by the noises. Specifically, DAPL extracts four features from sensor data: 1) lateral phone movement trajectory (i.e., left-to-right or right-to-left movement inside a car), 2) front/back phone movement trajectory, 3) BT RSSI variations, and 4) variations of magnetic-field magnitude. Since DAPL only utilizes the beacons broadcast by in-car BT, there is no need to establish a connection. Specifically, the feature of BT is used to determine whether the movement is pure front/back direction or involves lateral movement while the magnetic field is used to determine whether the movement is from front to back or back to front.

Note that even though DAPL utilizes similar sensor readings as prior work, DAPL does not utilize them in the same way. For BT RSSI, while Ho *et al.* [22] compared the absolute value of RSSIs from different anchors, DAPL captures the variation of BT RSSI when the phone is moving. While Park *et al.* [20] utilized the transient fluctuation (i.e., high-frequency variation due to status change of the engine) from the event

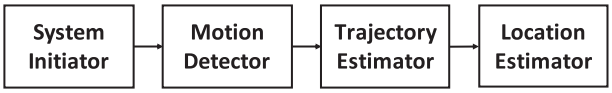


Fig. 2. System block diagram of DAPL.

of starting engine, DAPL utilizes the static/low-frequency components generated by in-car components to capture the feature of phone movement (i.e., the change of readings are due to the phone movement). Therefore, unless specified otherwise, the data considered in DAPL have already gone through low-pass filters.

To achieve the aforementioned functionalities and to facilitate its deployment in practice, DAPL's operation is broken down into four basic tasks, each of which is performed by the corresponding component. As shown in Fig. 2, DAPL is composed of 4 major components — System Initiator, Motion Detector, Trajectory Estimator and Location Estimator. First, DAPL utilizes System Initiator to monitor whether the phone has entered a given car, and start the location tracking process once a phone is determined to be inside the car (Section 3.2). Second, DAPL determines whether or not the phone is moving within the car by Motion Detector. Motion Detector identifies the start and the end of each movement and passes that information to the subsequent component (Section 3.3). Third, DAPL calculates the phone's movement trajectory utilizing Trajectory Estimator (Section 3.4). Finally, DAPL utilizes Location Estimator to determine the phone's location after the movement based on the extracted information (Section 3.5).

3.2 System Initiator

System Initiator determines whether or not the phone enters a car and obtains the initial phone location based on detecting the motion of entering a car [20]. If the phone enters a car, System Initiator will then activate Motion Detector to track the phone location inside the car. Otherwise, it will keep monitoring if the phone enters a car.

3.3 Motion Detector (MD)

The first step of phone localization is to determine whether or not the phone is moving inside the car or just moving along with the car. We propose Motion Detector to determine the start and end times of each phone's movement inside the car based on its angular speed and Magnetic-Field Changing Rate (MFCR). While the phone movement will inevitably induce both angular speed and acceleration, angular speed is chosen over acceleration because the latter can be continuously affected by the vehicle's motion, while the former is much more stable when the vehicle is moving on a straight road segment. Furthermore, angular speed will not only capture the phone's movement, but also the user's motion of picking up the phone prior to the phone's movement. Based on our preliminary testing, the angular speed of the latter is usually $>100\%$ greater than the angular speed caused by the natural vehicle vibration. MFCR is used as another indicator since the magnetic-field magnitude varies with the position [40], [41]. The magnetic-field magnitude differs between seats because of the location of mechanical and electrical components inside the car that induce magnetic field. In particular, there are more such components around the driver seat, and hence the driver seat tends to have the strongest magnetic field. See Section 3.5 for more discussion on the magnetic field inside a car.

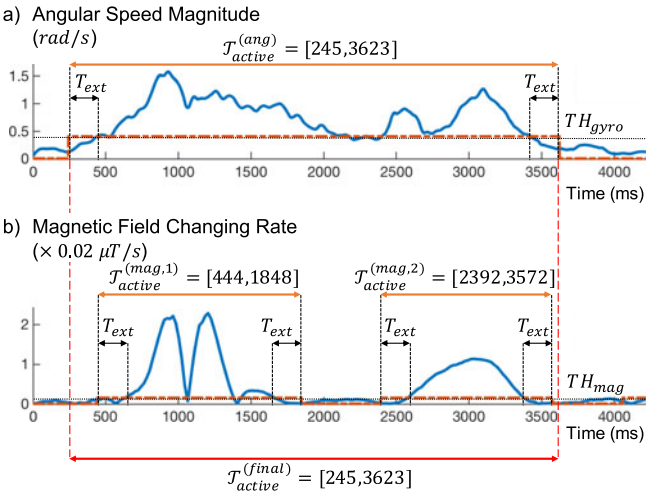


Fig. 3. Example of detecting an active period.

Let *active period* $\mathcal{T}_{active}$ be the time duration of a phone movement. An active/movement period is then determined by the following basic steps:

Step 1. Mark the recorded data at time t as *active* if the angular speed exceeds a parameter TH_{gyro} and extend the *active* time to $[t - T_{ext}, t + T_{ext}]$, where T_{ext} is a small time window used for phone's trajectory estimation (Section 3.4).

Step 2. Mark the recorded data at time t as active if MFCR exceeds a parameter TH_{mag} and extend the active time to $[t - T_{ext}, t + T_{ext}]$.

Step 3. Combine the overlapped active periods and output $[t_{act,s}, t_{act,e}]$ as the detected active period. Note that an active period will be reported only when there is an overlap between the active periods detected from Steps 1 and 2.

Fig. 3 shows an example of detecting an active period; the blue solid curve in Fig. 3a is the angular speed magnitude and the black dotted line is TH_{gyro}. Based on Step 1, the original active period starts at 445 ms and ends at 3423 ms, i.e., $\mathcal{T}_{active}^{(ang)} = [445, 3423]$. In this example, $T_{ext} = 200$ms, so $\mathcal{T}_{active}^{(ang)}$ is extended to $\mathcal{T}_{active}^{(ang)} = [245, 3623]$. The use of the extension time T_{ext} will be discussed more in Section 3.4. Similarly, in Step 2 (Fig. 3b), Motion Detector captures 2 active periods: $\mathcal{T}_{active}^{(mag,1)} = [444, 1848]$ and $\mathcal{T}_{active}^{(mag,2)} = [2392, 3572]$. Finally, Motion Detector combines the overlapped active periods and outputs the final result $\mathcal{T}_{active}^{(final)} = [245, 3623]$. Motion Detector will then pass the detected active period $\mathcal{T}_{active}^{(final)} = [245, 3623]$ to Trajectory Estimator. Selection of parameters will be discussed in Section 4.2.

To exclude the false-positive detection due to a vehicle's turn, Motion Detector may also determine if the angular speed and the car location data $\mathbf{X}_c(t)$ obtained from the location API of the phone OS (e.g., with GPS/WiFi) match the "signature" of a vehicle's turn [42]. If yes, it will discard that detection. Otherwise, Motion Detector will pass the detected phone movement to Trajectory Estimator.

3.4 Trajectory Estimator (TrajEst)

We propose TrajEst (Fig. 4) to calculate the lateral and front/back trajectories based on the phone's acceleration $\mathbf{a}_{[p]}(t)$ and

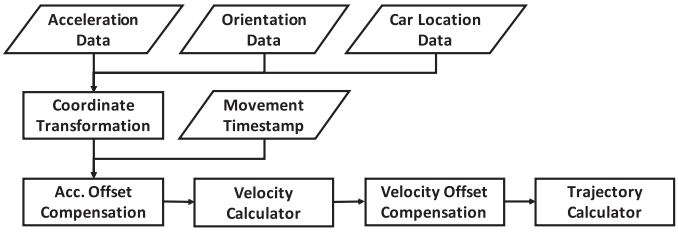


Fig. 4. Block diagram of Trajectory Estimator.

orientation $\Theta(t)$, the car location $\mathbf{X}_c(t)$ obtained from OS API, and the active period $\mathcal{T}_{active} = [t_{act,s}, t_{act,e}]$. Note that the goal of TrajEst is to capture the direction and pattern of the movement, but not necessarily to obtain the accurate estimation of phone displacement. To capture the phone movement inside a car, TrajEst needs to obtain $\mathbf{a}_{[c]}(t)$, the *relative acceleration* between the phone and the car. However, the acceleration $\mathbf{a}_{[p]}(t)$ recorded by the phone is represented in the Phone Body Coordinates (PBC) and its magnitude is the absolute value measured in an inertial coordinate system. Coordinate transformation and offset compensation (i.e., removing the acceleration/motion "offset" caused by car movement) must be performed to derive $\mathbf{a}_{[c]}(t)$ from $\mathbf{a}_{[p]}(t)$.

3.4.1 Coordinate Transformation

To obtain the movement trajectory $\mathbf{X}(t)$ from the acceleration $\mathbf{a}_{[p]}(t)$ recorded by the phone, $\mathbf{a}_{[p]}(t)$ needs to be rotated in the same direction as the Car Body Coordinates (CBC) and then subtract the car acceleration. Let $\mathbf{a}_{[p]}(t)$ denote the acceleration measured in PBC, $\mathbf{a}_{[g]}(t)$ be the acceleration in Geo-Coordinates (GC), and $\mathbf{a}_{[c]}(t)$ as the acceleration in CBC at time t. TrajEst rotates $\mathbf{a}_{[p]}(t)$ from PBC to GC by applying the rotation matrix R_{p2g} derived from the phone's orientation $\Theta(t)$. TrajEst then rotates it again to CBC by applying $R_{g2c}(t)$, which is derived from the car's traveling direction using car location data $\mathbf{X}_c(t)$, yielding

$$\mathbf{a}_{[c]}(t) = R_{g2c}(t)\ R_{p2g}(t)\ \mathbf{a}_{[p]}(t) - \mathbf{a}_{car[c]}(t) \quad (1)$$

$$= R_{g2c}(t)\ \mathbf{a}_{[g]}(t) - \mathbf{a}_{car[c]}(t) \quad (2)$$

$$= \mathbf{a}^*_{[c]}(t) - \mathbf{a}_{car[c]}(t), \quad (3)$$

where $\mathbf{a}^*_{[c]}(t)$ is the acceleration measurements of the phone represented in CBC direction.

Normally, the rotation matrix $R_{p2g}(t)$ can be obtained directly from the phone OSes. However, since the phone OSes, such as Android, use transient acceleration and magnetic field readings to estimate the phone's orientation in geo coordinates, $R_{p2g}(t)$ will not be accurate during the phone movement. Therefore, DAPL uses the R_{p2g} provided from the phone OS only for initial coordinate transformation and estimates the following rotation matrices based on the angular speed collected by gyroscope based on [43], a well-known phone-orientation estimation scheme (for PBC-to-GC conversion) in the case of phone movement. Note that the trajectory estimation error caused by the car acceleration is bounded by $2d\sin(0.5\tan^{-1}(a_{car}/g))$, where d is the actual phone displacement and g is the gravitational acceleration. Since the car acceleration is shown to be less than or equal to 1 m/s^2 for 92.67% of the time,[3] the trajectory estimation error caused by the car acceleration will be bounded by 0.0508m, 0.1016m, and 0.2033m when d = 0.5, 1, 2m, respectively. We refer readers to Appendices[4] for more details.

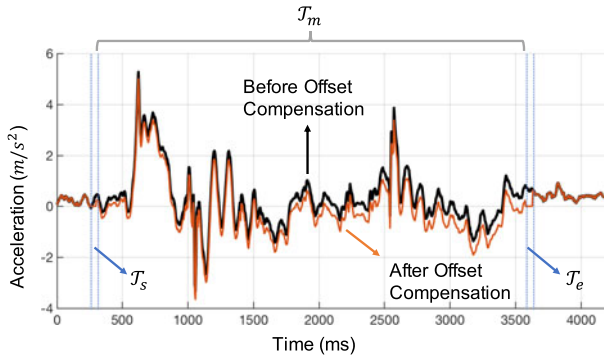


Fig. 5. This figure shows the time segments described in Eq. (6) and demonstrates the application of offset compensation to front/back direction acceleration.

3.4.2 Compensation of Movement-Time Offset

After the coordinate transformation, TrajEst performs offset compensation (i.e., removing the effects caused by car motion) to obtain $\mathbf{a}_{[c]}(t)$ and then acceleration integration to estimate the moving trajectories. DAPL exploits the fact that every movement can be partitioned into one or more small consecutive movements with the pattern of *stay still* → *move* → *stay still*. We use this movement characteristic to estimate $\mathbf{a}_{car[c]}(t)$ based on the fact that active periods (i.e., the duration of the movements) are usually short (i.e., average of 3.02s according to our experimental findings), and therefore, the car's acceleration is estimated with a linear approximation. First, we estimate $\mathbf{a}_{car[c],s}$, the car's acceleration at the beginning of an active period, and $\mathbf{a}_{car[c],e}$, the acceleration at the end of the active period. Second, we use an interpolation to estimate the car's acceleration for the rest of active period. Recall that Motion Detector extends the start and the end of an active period by T_{ext} in Section 3.3. Let $\mathcal{T}_s = [t_{act,s}, t_{act,s} + T_{ext}]$ and $\mathcal{T}_e = [t_{act,e} - T_{ext}, t_{act,e}]$ be the extended period at the start and the end of $\mathcal{T}_{active}$, respectively, and $\mathcal{T}_m = [t_{act,s} + T_{ext}, t_{act,e} - T_{ext}]$ be the period in between $\mathcal{T}_s$ and $\mathcal{T}_e$ (Fig. 5). Then, $\mathbf{a}_{car[c],s}$ and $\mathbf{a}_{car[c],e}$ are estimated by calculating the mean of $\mathbf{a}^*_{[c]}$ in $\mathcal{T}_s$ and $\mathcal{T}_e$, respectively.

$$\mathbf{a}_{car[c],s} = mean(\mathbf{a}^*_{[c]}(\mathcal{T}_s)); \quad (4)$$

$$\mathbf{a}_{car[c],e} = mean(\mathbf{a}^*_{[c]}(\mathcal{T}_e)). \quad (5)$$

$\mathbf{a}_{car[c]}(t)$ is then calculated by the following equation:

$$\mathbf{a}_{car[c]}(t) = \begin{cases} \mathbf{a}^*_{[c]}(t), & \text{if } t \in \mathcal{T}_s \text{ or } t \in \mathcal{T}_e \\ \frac{(t_{act,e}-T_{ext})-t}{|\mathcal{T}_{active}|-2T_{ext}}\ \mathbf{a}_{car[c],s} + & \\ \quad \frac{t-(t_{act,s}+T_{ext})}{|\mathcal{T}_{active}|-2T_{ext}}\ \mathbf{a}_{car[c],e}, & \text{if } t \in \mathcal{T}_m \end{cases} \quad (6)$$

3. The statistics derived from 110 traces in Safety Pilot open dataset [44].

4. Link to DAPL's Supplementary Materials. https://doi.ieeecomputersociety.org/10.1109/TMC.2022.3141019

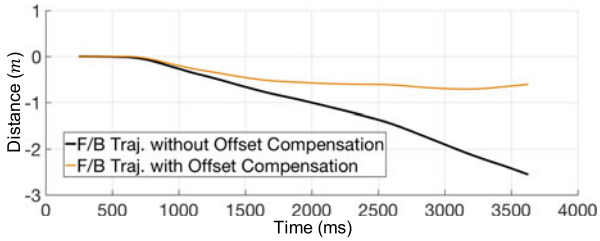


Fig. 6. Comparison between front/back trajectory estimations with and without the offset compensation.

Fig. 5 shows an example of applying the offset compensation to front/back direction acceleration, where the black thick curve is the raw acceleration and the orange thin curve is the acceleration after applying the offset compensation.

After obtaining $\mathbf{a}_{[c]}(t)$, we can perform integration to get the velocity $\mathbf{v}^*_{[c]}(t)$. Similarly, based on the special pattern of phone movement, we perform the offset compensation to obtain $\mathbf{v}_{[c]}(t)$=$\mathbf{v}^*_{[c]}(t) - \mathbf{v}_{car[c]}(t)$. Finally, we can obtain $\mathbf{X}(t)$ by integrating $\mathbf{v}_{[c]}(t)$. Fig. 6 compares the final front/back trajectory estimations with and without the offset compensation. Without offset compensation, the phone is estimated to move 2.55m while the actual movement is about 0.8m.

3.5 Location Estimator (LocEst)

We propose the use of 4 features to enable noise-resilient in-car phone localization. Location Estimator takes the results from Trajectory Estimator as input and determines whether or not the phone has been moved to a different seat. Basically, LocEst finds the movement that matches best with the phone movement. Fig. 7 shows the block diagram of LocEst that consists of Trend Extractor, Case Constructor, and Case Matcher. Trend Extractor derives the trend of the results from previous components and passes the derived information to Case Matcher, which then matches the current movement with one of the rows (cases) in the table of movement features created by Case Constructor and outputs the final result.

3.5.1 Trend Extractor

Trend Extractor determines the data trends of lateral trajectory, front/back trajectory, Bluetooth RSSI, and magnetic-field magnitude from the processed data. Each data trend is described by a quadratic polynomial function $f(t) = a + bt + ct^2$ and active period $\mathcal{T}_{active}$. Coefficients a, b, and c are determined by applying the ridge regression [45] to the feature vectors generated from the processed data $\{(t_i, x_i)\}$. The rationale behind the choice of a quadratic polynomial function is that it can describe the data variations/trends that DAPL needs (Section 3.5.2) and its coefficients can be obtained efficiently using a closed-form solution [45]. Fig. 8a shows an example of extracting the trend of front/back trajectory. The solid curve is the data input from TrajEst and the dashed curve is the regression result $f(\mathcal{T}_{active})$.

3.5.2 Case Constructor

The movement feature table is constructed by Case Constructor. Every row in the movement feature table is a possible movement case. As shown in Fig. 1, we divide the car into four zones Z_i, $i = 1, \ldots, 4$, where Z_1 is the driver seat and Z_2–Z_4 are passenger seats. Therefore, there are a total of

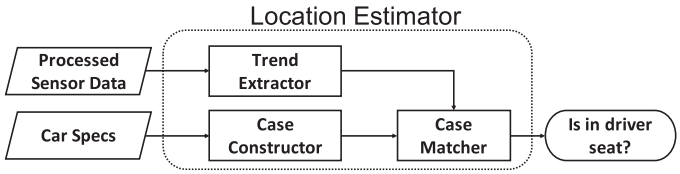


Fig. 7. Block diagram of Location Estimator.

16 possible movements between zones. Case Constructor creates a feature table with parameters, depending on the car specification with four columns for each row: 1) magnetic-field magnitude, 2) Bluetooth RSSI, 3) lateral trajectory, and 4) front/back trajectory. Because most car-makers have already embedded the model name of the car (e.g., [46], [47], [48]) or that of the infotainment system (e.g., [49], [50], [51]) in the broadcast BT signal, DAPL can not only use this embedded info to identify the BT signal to be tracked but also obtain the physical dimension of the car.

Before introducing the possible values of each entry in the movement feature table, we first define some notations. For a given trend $f(\mathcal{T}_{active})$ as the example in Fig. 8a, let $t_{0.25}$ and $t_{0.75}$ be the timestamps of 25 and 75 percentile of the movement, which are marked with reverse triangles, and t_v be the timestamp where the vertex of $f(\mathcal{T}_{active})$ is located. Let x^*_s be the initial value of $f(\mathcal{T}_{active})$ and x^*_e be the ending value of $f(\mathcal{T}_{active})$. Finally, let x^*_{max} and x^*_{min} denote the maximum and minimum values in $f(\mathcal{T}_{active})$. The value of each entry can then be one of the following values:

- Same level ($=$): if $|x^*_s - x^*_e| < TH$;
- Not same level ($\neq$): if is not the same level;
- Concave ($\bigcap$): if $t_v \in [t_{0.25}, t_{0.75}]$, $a < 0$, and is $\neq$;
- Convex ($\bigcup$): if $t_v \in [t_{0.25}, t_{0.75}]$, $a > 0$, and is $\neq$;
- Increasing ($\nearrow$): if is not $\bigcap$ or $\bigcup$, and $x^*_s < x^*_e$;
- Decreasing ($\searrow$): if is not $\bigcap$ or $\bigcup$, and $x^*_s > x^*_e$.

The parameters for determining the same level is different for each column. Let TH^{small}_{mag}, TH^{small}_{BT}, TH_{lat}, and $TH_{f/b}$ be the parameters. We will discuss how to obtain these parameters automatically in Section 4.2. Here we demonstrate a typical movement feature table (Table 2), where the BT transceiver is embedded in the infotainment system. Note that most BT transceivers are built in infotainment system modules located between the driver seat and the front passenger seat, and are automatically turned on when the engine is started [46], [47], [48], [49], [50], [51]. Therefore, if a phone moves laterally, the BT RSSI will have a $\bigcap$ trend (Fig. 8d), else it will have a $\nearrow$ or $\searrow$ trend. The magnetic field entries are determined using the fact that the driver seat has the strongest magnetic field in gasoline-powered vehicles and the front row seats have a stronger magnetic field than back-row seats. This fact is confirmed by our measurements and also by [41]. Since the values in the entries can be determined based on the aforementioned rules and simple laws of physics, we will omit the detail of every entry. While Table 2 shows the features of common four-seat cars, we refer readers to Appendices for more discussions on i) how to adapt DAPL to different car models (e.g., electric vehicles) based on their in-car environment [41], [52], [53], [54] and ii) DAPL's deployment.

3.5.3 Case Matcher

Case Matcher marks the entries to have 1 in the feature table if the entry matches the corresponding trend; 0 otherwise.

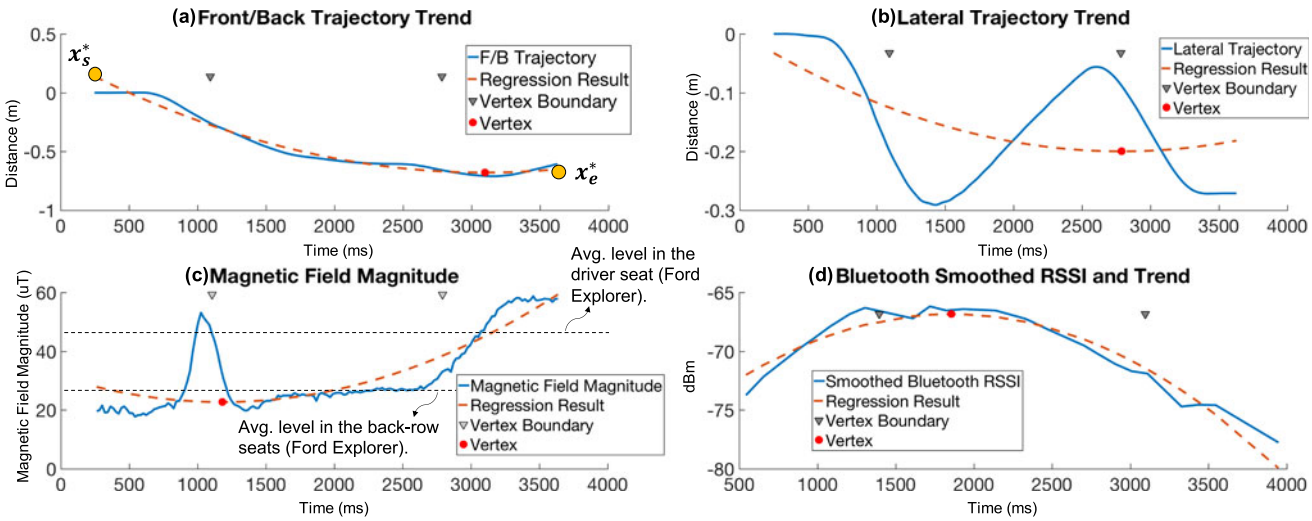


Fig. 8. This figure shows an example when the phone was moved from Z_4 to Z_1, where the vertex boundaries ($t_{0.25}$ and $t_{0.75}$) are chosen to capture the major trend of the value variation with greater than majority (50%) of movement time.

After marking all the entries, Case Matcher takes the sum of each row (case) and finds the one with the highest score to be the final output. Let us illustrate how Case Matcher works with an example when the phone was moved from Z_4 to Z_1. Fig. 8c shows the magnitude of magnetic field and its increasing trend. Fig. 8d shows the Bluetooth RSSI and its concave trend. Fig. 8b shows the lateral trajectory and its decreasing trend. Finally, Fig. 8a shows the front/back trajectory and its decreasing trend. This trace is a perfect match of the case of $Z_4 \rightarrow Z_1$ movement, and therefore, Case Matcher concludes that the phone was moved from the right rear seat to the driver seat.

3.6 Energy-Saving Mechanism

Energy consumption of a phone is an important issue that influences user experience. In theory, we can implement DAPL as an "always-on" service with all the sensors operating at the highest sampling rate to achieve the best performance, but doing so will incur unnecessary and excessive energy consumption. To address this issue, we introduce a three-phase operating mechanism. Fig. 9 shows the operating states of DAPL: OFF, SLEEP and ACTIVE. DAPL stays in OFF mode when the user is not inside a car. After the user enters the car, DAPL enters SLEEP mode with only gyroscope turned on and operating at a low sampling rate only for detecting the start of a movement. In the current implementation of DAPL, the sampling rate is set to 10Hz. The start of a movement is detected by checking if the angular speed magnitude exceeds TH^*_{gyro}. To ensure DAPL will have enough time for the compensation of offset, TH^*_{gyro} is designed to be smaller than the parameter TH_{gyro} used in Motion Detector. We will discuss more about the effect of TH^*_{gyro} in Section 5.

Upon detection of the start of a movement, DAPL enters ACTIVE mode where all the required sensors are operating at a high sampling rate. The sampling rate is set to Android's pre-defined sampling rate "SENSOR_DELAY_FASTEST" in the current implementation. When the movement ends, DAPL returns to SLEEP mode. If DAPL determines that the phone is no longer in the car, it will enter OFF mode. DAPL determines the phone not to be inside the car when it does not receive any BT signal from the car. Since there already exist ways, to determine how the phone entered the car [20], we will focus on the implementation and evaluation of SLEEP and ACTIVE modes on a DAPL prototype.

TABLE 2
A Table of Typical Movement Features

Movement	Mag. Field	BT RSSI	Lat. Traj.	F/B Traj.
$Z_i \rightarrow Z_i$	$=$	$=$	$=$	$=$
$Z_1 \rightarrow Z_2$	$\searrow$	$\bigcap$	$\nearrow$	$=$
$Z_2 \rightarrow Z_1$	$\nearrow$	$\bigcap(\nearrow)_{iii}$	$\searrow$	$=$
$Z_3 \rightarrow Z_4$	$=(\nearrow)_i$	$\bigcap$	$\nearrow$	$=(\bigcap / \bigcup)_{iv}$
$Z_4 \rightarrow Z_3$	$=(\searrow)_{ii}$	$\bigcap$	$\searrow$	$=(\bigcap / \bigcup)_v$
$Z_1 \rightarrow Z_3$	$\searrow$	$\searrow$	$=$	$\nearrow$
$Z_3 \rightarrow Z_1$	$\nearrow$	$\nearrow$	$=$	$\searrow$
$Z_2 \rightarrow Z_4$	$\neq$	$\searrow$	$=$	$\nearrow$
$Z_4 \rightarrow Z_2$	$\neq$	$\nearrow$	$=$	$\searrow$
$Z_1 \rightarrow Z_4$	$\searrow$	$\bigcap$	$\nearrow$	$\nearrow$
$Z_4 \rightarrow Z_1$	$\nearrow$	$\bigcap$	$\searrow$	$\searrow$
$Z_2 \rightarrow Z_3$	$\neq$	$\bigcap$	$\searrow$	$\nearrow$
$Z_3 \rightarrow Z_2$	$\neq$	$\bigcap$	$\nearrow$	$\searrow$

The symbols in the parentheses are some features that we included in the evaluation due to practical usage scenarios. The adjustments i and ii are made because the right back-row seat is found to have a stronger magnitude of magnetic field according to our measurements. The inclusion of iii reflects the fact that drivers tend to put their phones in a phone stand right in front of the infotainment system. The adjustments of iv and v are added because there is more room to move the phone in the back-row seats.

3.7 Refinements

3.7.1 Use of Last Known Location

Since the detection of entering a car [20] also provides the information of the phone's initial location $Z^{(\ell)}$, DAPL can

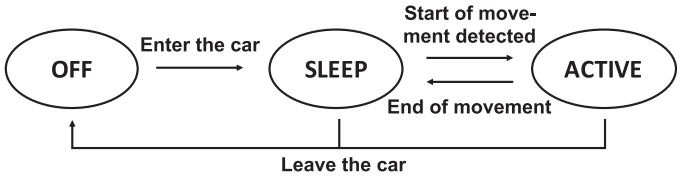


Fig. 9. Operating states of DAPL.

utilize this information as the last known location and consider only those cases starting from $Z^{(\ell)}$ to enhance localization accuracy. If all the cases originating from $Z^{(\ell)}$ have less than 2 matching entries, then Case Matcher will conclude that $Z^{(\ell)}$ is incorrect, and thus consider all other cases to find the best match.

3.7.2 Safety-First Choice

Note that it is safe to assume the phone in the driver seat even if it were not, because this misclassification can only lead to unavailability of certain functions on the phone. In contrast, it would be dangerous if the phone is in the driver seat and DAPL fails to detect its true location (i.e., the driver has full access to his phone even when he is not allowed to use it). So, to ensure safety, DAPL will add 1 point to the scenarios that end in the driver seat (Z_1) if the magnetic field strength rises by twice TH_{mag}^{small} as it is a strong indication that the phone may have entered the car to the driver seat. This is based on our preliminary experiments in which the magnetic field measurement is found more reliable than other sensor measurements. Thus, we use the extra 1 point to increase the weight of this feature when this event occurs. Also, we will add an extra 0.5 point to the scenarios that end in the driver seat (Z_1). That way, DAPL will always determine the phone to be in the driver seat if there is a tie in the case matching without violating the partial ordering of other scenarios. Due to this Safety-First-Choice, DAPL will always assume that the phone is in the driver seat if DAPL cannot tell whether or not the movement ends in the driver seat. This may lead to false positives (which is safe but may be annoying). Therefore, DAPL also accounts for strong signatures that the phone has entered a passenger seat, or specifically, a back-row seat. Since back-row seats are farther away from the sources of magnetic field and BT signals, their values tend to be lower than those in the driver/front-row seats. DAPL adds 1 point to those cases ending in a back-row seat if the values of magnetic field or BT RSSI readings are smaller than TH_{mag}^{d} and TH_{BT}^{d}, respectively. We will discuss how these parameters are chosen in Section 4.2.

4 Performance Evaluation

4.1 Experimental Settings

We conducted experiments with Huawei Nexus 6P, LG Nexus 5X, and Sony Z1 on 3 different cars: Ford Explorer (SUV), Honda Fit (sedan), and Toyota Corolla (sedan). As shown in Fig. 12, we tested 8 scenarios each of which represents a commonly seen in-car phone movement, such as movement between seats or picking up the phone from the phone stand or cup holder. In Scenario 1, the phone is moved between the front passenger seat and the driver seat. This is the case when the driver moves his phone from the phone stand or a bag located in the front passenger seat, or the passenger passes the phone to the driver and in their reverse directions. In Scenario 2, the driver places his phone on the phone stand in the left end of front passenger seat and retrieves it back to the driver seat. In Scenarios 3 and 4, a passenger in the back row hands his phone to the driver or the driver hands it back to the passenger. In Scenario 5, the user moves the phone within the vicinity of the seat, such as the driver (or a passenger) retrieves/puts the phone from/to the dashboard, the cup holder, the phone stand, or his pocket. We do not evaluate the condition in which the driver directly puts his phone on the phone stand right after entering the car because this can be prevented by utilizing the initial location. In Scenarios 6–8, a passenger takes/puts the phone from/to another passenger seat. See Fig. 1 for detailed zone definition.

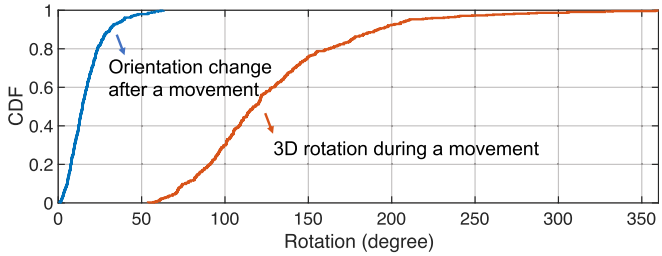


Fig. 10. This figure shows the CDFs of i) change of phone orientation after a movement represented by a (minimum) quaternion rotation and ii) the total (3D) phone rotation during a movement in the test cases.

We have conducted experiments while driving around urban areas (including our campus) and freeways. The routes we took in urban area include intersections with traffic lights, stop signs, and speed bumps, The tested road conditions include flat and hilly roads with up to 6.7% grade [55]. We honored the speed limit of 25–35 mph, and maintained the speed of cars close to the speed limit, depending on the traffic. We conducted experiments during 9–11AM and 4–6PM under different weather conditions, including snowing, raining, and sunny. Note that our choice of testing area/routes and time in urban area is to capture the condition in which distracted driving will be most likely to cause accidents since the drivers must focus on the road due to busy traffic and pedestrians crossing roads constantly [56], [57]. While we also conducted experiments on freeway ($\leq$70 mph or 112.65 kph), only the cases without involving drivers were tested on high-speed freeways to ensure the safety of participants. During the experiments, at least one person in the car was always monitoring surrounding environment, and no experiment was conducted when there were other cars close by to ensure the safety of the participants. Note that besides the above safety measure, we did *not* instruct drivers how to control the vehicles, such as maintain a constant speed, or change the way of moving/handling their phones to ensure the realism of test cases.

For each scenario, we tested common use-cases in which the phones can be rotated along with multiple axes or even flipped (i.e., 360° rotations) during the movement and can be moved with different initial/resulting phone orientations (i.e., how the phone is placed). Fig. 10 shows the statistics of the phone orientation change and overall (3D) rotation in the test cases and Fig. 11 shows the statistics of the average phone moving speed. Also, the traveling distance and bearing change of the car during a phone movement can be up to 311.36ft (94.9m) and 174° (i.e., a "U" turn), respectively. The time duration of a phone movement can be from $<$1s to 10s. Since the experiments show consistent results for different conditions and it is infeasible to show all the detailed settings given the space limit, we do not present them individually in this paper. See Appendix-C.1 for more test-case information, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TMC.2022.3141019.

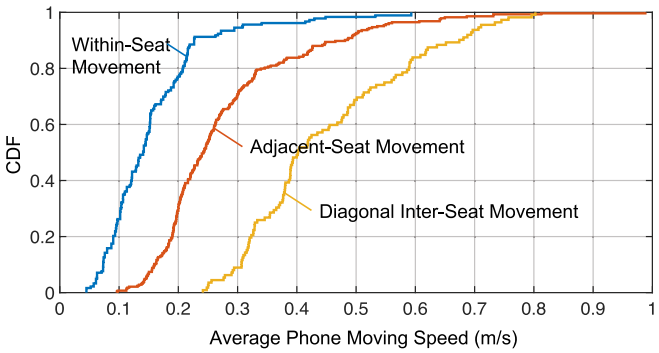


Fig. 11. This figure shows the statics of the average phone moving speeds (i.e., the displacement of the phone on the horizontal plane divided by the movement time).

We use accuracy, false positive rate, and false negative rate to evaluate DAPL. Accuracy is defined as the ratio of correctly identified cases to the total number of cases

$$\text{Accuracy} = (\#\text{ of Correct Cases})/(\text{Total }\#\text{ of Cases}). \tag{7}$$

A false positive occurs when the phone is not in the driver seat, but DAPL determines it to be in the driver seat. The false positive rate (FPR) is thus computed as

$$\text{FPR} = (\#\text{ of False Positive Cases})/(\text{Total }\#\text{ of Cases}). \tag{8}$$

A false negative occurs when the phone is in the driver seat, but DAPL determines it to be in a passenger seat. The false negative rate (FNR) is thus computed as

$$\text{FNR} = (\#\text{ of False Negative Cases})/(\text{Total }\#\text{ of Cases}). \tag{9}$$

4.2 Parameter Training

Motion Detector (MD) and Location Estimator (LocEst) utilize parameters for their detection and the training process is very simple. The idea of parameter training is to capture the sensor readings while the phone stays still in the driver seat. Based on our preliminary testing with 150 moving and 4 stay-still traces, a 5-min drive or a trip of users' regular route suffices for automatic selection of the parameters. DAPL will determine the parameter as follows.

TABLE 3
Settings Used in the Evaluation

Param.	Value	Used in	Param.	Value	Used in
TH^*_{gyro}	0.2 rad/s	MD	$TH_{f/b}$	0.5 m	LocEst
TH_{gyro}	0.4 rad/s	MD	TH^d_{mag}	40 μT	LocEst
TH_{mag}	7.5 μT/s	MD	TH^{small}_{mag}	25 μT	LocEst
T_{ext}	200 ms	MD	TH^d_{BT}	-80 dBm	LocEst
TH_{lat}	0.5 m	LocEst	TH^{small}_{BT}	5 dBm	LocEst

We first look at the parameters in MD. TH_{gyro} and TH_{mag} should be set so as not to trigger MD if the phone stays still in a moving car. In such a case, the gyroscope will capture both the motion/vibration caused by vehicle components and the road condition. Since the high frequency motion of the former can be eliminated by having the data go through a low pass filter, TH_{gyro} can be determined by identifying the angular speed caused by the road condition. So, TH_{gyro} is set to the recorded maximum angular speed magnitude (after low-pass filtering) when the phone stays still. Likewise, TH_{mag} can be set. TH^*_{gyro} can be chosen differently according to developers' choices, and the effect of choosing different values will be discussed in Section 5.

We now look at the parameters in LocEst. TH^{small}_{mag} and TH^{small}_{BT} are set to the maximum variation of the corresponding sensor data recorded when the phone stays still in a moving car. TH^d_{mag} and TH^d_{BT} were chosen to be the minimum of sensor readings in the driver seat. TH_{lat} and $TH_{f/b}$ were determined by the dimension of the car. $TH_{f/b}$ was set to a value smaller than a half of the distance between the front and the back seats, while TH_{lat} was set to a value smaller than a half of the distance between the driver seat and the front passenger seat.

The parameters presented in Table 3 are the settings used in the performance evaluation (data collected from Nexus 5X with a 5-min trace) and we did not change their settings for different cars or phones during the basic scenario evaluation. Note that TH_{lat} and $TH_{f/b}$ used in our evaluation are based on dimensions of small sedans and they can be directly applied to larger cars. For the regular sedan and SUV, these thresholds can be set to the values provided

Scenario 1

Car	←	→	FPR	FNR	ACC
SUV	39	35	0	4.05	95.95
Sedan	12	10	0	9.09	90.91
Subtotal	51	45	0	5.21	94.79

Scenario 2

Car	←	→	FPR	FNR	ACC
SUV	26	25	0	3.92	96.08
Sedan	49	23	0	6.94	93.06
Subtotal	75	48	0	5.69	94.31

Scenario 6

Car	↙	↗	FPR	ACC
SUV	12	13	0	100
Sedan	10	10	10	90.00
Subtotal	22	23	4.44	95.56

Scenario 7

Car	↑	↓	FPR	ACC
SUV	12	16	28.67	71.43
Sedan	11	12	4.35	95.65
Subtotal	23	28	17.65	82.35

Scenario 5 (Within Seat)

Car	↺	FPR	FNR	ACC
SUV	49	0	0	100
Sedan	11	0	0	100
Subtotal	60	0	0	100

Scenario 3

Car	↖	↘	FPR	FNR	ACC
SUV	24	29	0	11.32	88.68
Sedan	7	7	0	14.29	85.71
Subtotal	31	36	0	11.94	88.06

Scenario 8

Car	←	→	FPR	ACC
SUV	26	23	20.41	79.59
Sedan	10	10	5.00	95.00
Subtotal	36	33	15.94	84.06

Cases with Small Movement

Car	↺	FPR	FNR	ACC
SUV	40	0	5.00	95.00
Sedan	52	0	5.77	94.23
Subtotal	92	0	5.43	94.57

Scenario 4

Car	↑	↓	FPR	FNR	ACC
SUV	26	30	0	7.14	92.86
Sedan	6	6	0	16.67	83.33
Subtotal	32	36	0	8.82	91.18

Scenario 1 – 5 Subtotal

Cases	FPR	FNR	ACC
414	0	6.28	93.72

Scenario 6 – 8 Subtotal

Cases	FPR	ACC
165	13.33	86.67

Total

Cases	FPR	FNR	ACC
579	3.80	4.49	91.71

Fig. 12. This figure shows the 8 testing scenarios and their evaluation results. The columns with arrows are the number of cases tested in each direction. FPR, FNR, and ACC are all represented in percentage (%).

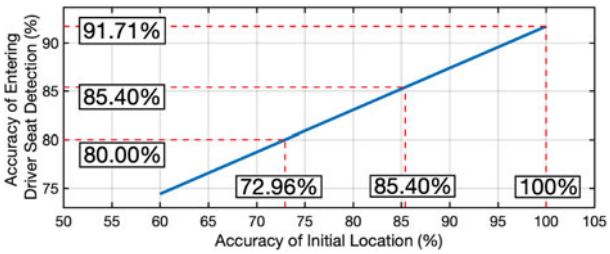


Fig. 13. Accuracy of entering driver seat detection based on different initial location accuracies.

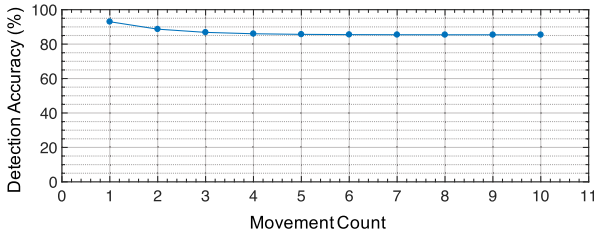


Fig. 14. This figure shows the (average) performance when the phone experiences multiple consecutive movements.

here unless there are some special modifications to the car. This suggestion is grounded on the fact that even when the training is performed on a bumpy road (as a worst-case scenario), we have not observed any significant performance variation with different phone-car combinations. If developers want to adjust the thresholds for better performance at runtime, they can do so by extracting the car model/make embedded in the BT signal as discussed in Section 3.5.2 and prompting a dialog to users to enable the parameter training process.

4.3 Results

4.3.1 Basic Scenarios

Fig. 12 summarizes our evaluation results. We have collected 579 test cases to evaluate DAPL. The accuracy of detecting movements between the driver seat and a passenger seat or moving within a specific seat (Scenarios 1–5) is 93.72% and that between passenger seats (Scenarios 6–8) is 86.67%. The overall accuracy of DAPL is 91.71%. As expected, Scenarios 1–5 have higher accuracy than Scenarios 6–8 thanks to the adoption of Safety-First-Choice policy. In Scenarios 1–6, the accuracies of SUVs are higher than those of sedan because the phone's movement distances are larger in SUVs and, therefore, the trends of movements are easier to capture. However, because phones in SUVs have more space to move, even within a seat, it can lead to more false positives while the phone is moving between passenger seats. Therefore, Scenarios 6 and 7 of SUV have lower accuracies than those of sedan.

Note that 92 of the test cases contain small movements ($\leq$ 20cm) within seats and the accuracies of those cases are 94.57%. This means that DAPL can achieve high accuracy even if the phone movements consist of several small movements when the phone is moving from one seat to another. For example, the driver takes his phone out of his pocket and then hands it to a passenger. Another example could be that when the passenger wants to hand the phone to the driver, the driver does not grab the phone immediately, but takes a few seconds to grab it, or vice versa.

4.3.2 Incorrect Initial Location

We also evaluate how DAPL can improve the existing solutions. Since DAPL utilizes the last known location for location estimation, we consider the condition under which the last known location is incorrect. We took the 189 traces (Scenarios 1-4) in which the phone was moved from a passenger seat to the driver seat and an incorrect initial position was used to evaluate DAPL's performance with wrong information. For every trace, we tested the condition in which DAPL used the other 2 passenger seats as the initial locations and we compute the expected performance under different initial location accuracies. Fig. 13 shows the relation between accuracy of detection of entering the driver seat and the accuracy of initial location. DAPL can still achieve good accuracy after a movement even if the detection of entering a car was incorrect. In particular, DAPL achieved 80% of detection accuracy for the detection of entering the driver seat even if the initial location was only 72.96% accurate. That is, DAPL is able to "correct" the result of detection of entering the driver seat when the initial phone location information was wrong.

4.3.3 A Sequence of Phone Movements

Let us consider the performance of DAPL after a sequence of inter-seat movements. Since DAPL's localization only depends on the last known location before the movement and the sensor readings of the movement, the average accuracy after all possible combinations of inter-seat movements can be derived from a Markov Chain analysis based on the performance of different single movements. Specifically, if DAPL has a correct initial location and does not calibrate between two movements, the expected (average) accuracy for detecting whether the phone moves to the driver seat is shown in Fig. 14. As mentioned in Section 4.3.2, since DAPL has the ability to "correct" the phone location after a movement even if the last known location is incorrect, it is able to reach a stable/equilibrium state to produce a $\sim$85% (average) detection accuracy after 5 movements.

4.3.4 Phone Movements in Driver's Clothing

We now consider the condition where the phone is kept in the driver's clothing and see if DAPL reports (incorrect) inter-seat movements due to his/her actions (e.g., applying the gas and brake pedals). Note that since smartphones are all equipped with proximity sensors, the movements in the driver's clothing can directly be filtered out if the phone is detected to be in a confined place. However, we still conduct experiments to see whether DAPL can function correctly without this mechanism. Specifically, we tested three most common scenarios where the phone is kept in i) the left pocket of the driver's pants, ii) the right pocket of the driver's pants, or iii) the pocket of the driver's upper body clothing (i.e., jacket and shirt) while driving in the urban area near our campus where the traffic was busy and the driver needs to constantly apply gas or brake pedal. (See Appendix C.1 for a test-case example, available in the online supplemental material.)

Table 4 shows the experimental results. One can observe that DAPL did detect movements in all three scenarios, and

TABLE 4
Detection Results When the Phone is in Driver's Clothing, Where a Detected Movement can be Filtered by Time, a Turn, or Both and the Only Incorrect Detection is Caused by the Vehicle Travelling on an Extremely Bumpy Road With Multiple Potholes

Phone Location	Time	Total Detected Movement	Filtered by Time (<1s)	Filtered by a Turn	Incorrect Detection
Left Pocket (Pants)	6.13′	4	4	4	0
Right Pocket (Pants)	6.43′	4	1	3	1
Upper Body (Jacket/Shirt)	6.99′	3	3	1	0

can still distinguish them (i.e., inner-seat movements) from inter-seat movements by determining whether the movement has a short movement time or the detected movement is caused by the car making a turn (Section 3.3). During our experiments, there is only one (incorrect) detection that DAPL determines the phone is having an inter-seat movement. However, this detection is not caused by the driver's movement but the car travelling on an extremely bumpy road, making the car shake horizontally (i.e., a rare case). In summary, even without utilizing the proximity sensor, DAPL is unlikely to have incorrect detection results in normal driving conditions when the phone stays still in a confined space (e.g., within a seat or in the driver's clothing).

4.3.5 Trajectory Trend Under Different Movement Speeds

Figs. 15 and 16 further show DAPL's capability of identifying the (matching) patterns/trends of horizontal and F/B phone movements, respectively, at different phone-movement speeds. Specifically, other than the lower performance when the phone has a F/B movement at 0.4–0.5 m/s speed, DAPL can consistently achieve $>70\%$ probability of identifying the movement pattern. Note that even though the above performance may seem to be underwhelming due to its reliance solely on smartphone sensors for trajectory identification, these movement patterns are still important to enhance DAPL's performance; next we will look into DAPL's performance when certain features are not available to DAPL.

4.3.6 Absence of Features

When DAPL utilizes only two features, the average detection accuracy of all combinations is 68.0% and that of three features is 81.3%. Specifically, DAPL can still achieve 65.28% and 82.56% accuracy without access to BT and magnetic field measurements, respectively. The former represents the condition that there is no built-in BT transceiver in the car. The latter represents the case when the vehicle is experiencing strong magnetic interference from the environment or in-car components. The accuracy degrades to 83.94% without the offset compensation and that without the Safety-First Choice is 80.31%, showing their importance to accuracy enhancement.

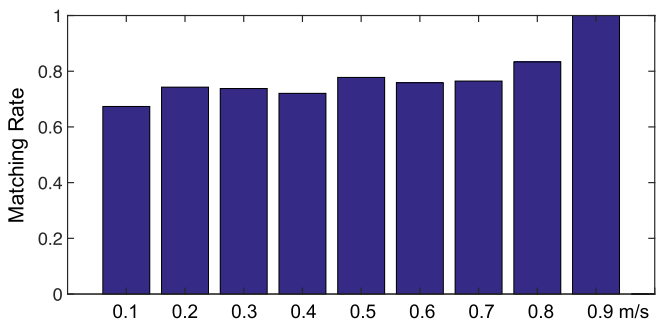


Fig. 15. The probability of identifying the matching horizontal movement pattern, where k in x-axis indicates that the (average) phone-movement speed lies in the range of $(k-0.1,\ k]$ m/s.

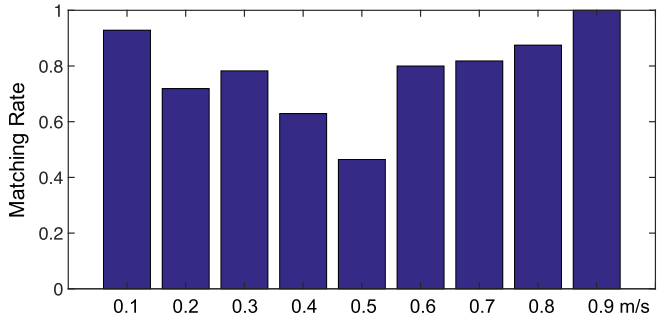


Fig. 16. The probability of identifying the matching F/B movement pattern, where k in x-axis indicates the (average) phone-movement speed lies in the range of $(k-0.1,\ k]$ m/s.

4.3.7 Comparison With Prior Work

While the built-in driving mode cannot determine where the phone is being used [32], [33], let us evaluate DAPL's performance in comparison with prior work with 93% initial location detection accuracy in [20]. DAPL is able to achieve 88+% accuracy even after phone movements, compared to [18] ([19]) that provides 87% (85%) accuracy for only initial location detection. While both utilize the built-in BT transceiver, DAPL is able to achieve comparable accuracy (91+%) as [26] without the requirement of upgrading the in-car audio system to a 4-channel output. Even though DAPL's performance falls short compared to those approaches with dedicated hardware, such as [23] (80-98+%, depending on the driving scenarios), [24] (90+%), [22] (95+%), and [25] (99+%), DAPL has the advantage of easy deployment and no requirement of installing additional dedicated hardware. On the other hand, even though [21] does not require additional hardware for its deployment and reports $>99\%$ accuracy, it is only designed to function when the phone stays still within the car (Section 2). As mentioned earlier, since our design goal is to reduce the occurrences of distracted driving, it is our belief that enabling wide adoption will be more effective than having a slightly better detection accuracy with less incentive for deployment. This is grounded on the fact that while the detection accuracy can only marginally increase (up to 3-12%) with additional hardware, the number of deployments or adoptions is likely to increase much more than that if DAPL is built-in in phone OSes or can be downloaded from Play/App store directly.

5 Energy Consumption

5.1 Evaluation Settings

Nexus 5X is used to evaluate DAPL's energy consumptions in SLEEP and ACTIVE modes. During the evaluation of

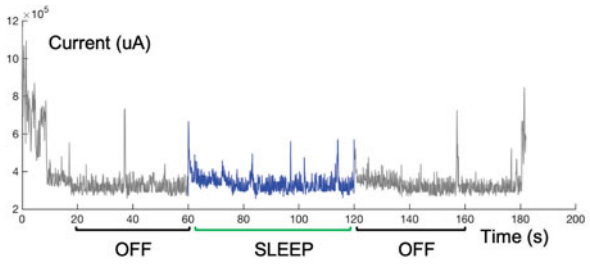


Fig. 17. Example of electric current in SLEEP mode.

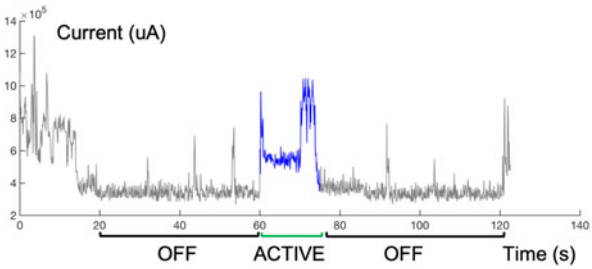


Fig. 18. Example of electric current in ACTIVE mode, where the first peak in ACTIVE mode is the result of sensor activation, the lower plane is the state for sensor data recording, and the peak in the end is contributed by the computation. Note that this current profile was recorded for illustration purpose. The average duration for DAPL in ACTIVE mode is 3.02s, which is much shorter than the case we demonstrated here. Therefore, the duration for sensor data recording and computation will also be much shorter in real-world case.

SLEEP mode, only gyroscope, WiFi and Bluetooth stay on, and both WiFi and Bluetooth are not connected to any other device. After measuring the energy overheads (i.e., the additional current drawn from the battery) of DAPL in SLEEP and ACTIVE modes, we evaluate the effectiveness of the proposed energy-saving mechanism by measuring the false negative rate (FNR_e) and the false positive rate (FPR_e) of entering ACTIVE mode. FNR_e is defined as the ratio of the time that the phone is moving while DAPL is still in SLEEP mode to the total time of the phone's moving: $FNR_e = \Sigma_i T^{(moving)}_{\text{SLEEP},i} / \Sigma_i T^{(moving)}_i$. FNR_e indicates the missed detection time when the phone is moving. Therefore, the higher FNR_e, the lower localization accuracy. Similarly, FPR_e is defined as the ratio of the time that the phone stays still, but DAPL is in ACTIVE mode, to the total time that the phone stays still: $FPR_e = \Sigma_i T^{(still)}_{\text{ACTIVE},i} / \Sigma_i T^{(still)}_i$. While FPR_e does not have direct influence on the localization accuracy, it indicates how much energy is wasted due to falsely entering ACTIVE mode. We utilize the same set of traces used in Section 4 to calculate FNR_e and collect additional 4 traces that the phones stayed still in the cars to calculate FPR_e.

5.2 Results

To assess the DAPL's energy overhead (OH) under common phone usage scenarios, we measured the difference between the discharge current with DAPL in SLEEP/ACTIVE mode and that with DAPL turned off (Figs. 17 and 18). Table 5 summarizes the results of energy consumption overhead. "(Screen) On" is the case when the phone screen is adjusted to 50% brightness and "(Screen) Off" is the case when the screen is turned off. We evaluate the energy overhead of sensor data collection only, indicating a lower bound of overhead, and the overall energy overhead of DAPL by considering both sensor data collection and computation, indicating an upper-bound of energy overhead. The average absolute OH of SLEEP and ACTIVE mode are 5 and 198mA, respectively. Compared to the energy consumption with the screen off, DAPL incurs 6.81% overhead in SLEEP mode and 207.02% overhead in ACTIVE mode.

TABLE 5
DAPL's Energy Overhead

Mode	SLEEP,On	ACTIVE,On	SLEEP,Off	ACTIVE,Off
Sensor Only	0.67%	36.58%	2.89%	158.92%
All	1.57%	66.50%	6.81%	207.02%

TABLE 6
DAPL's Tradeoff Between Energy Consumption and Localization Accuracy

R_{act}	FNR_e	FPR_e	En. OH	En. Saved	ACC
0.50	2.7%	23.3%	2.73%	98.68%	91.71%
0.75	5.5%	2.5%	0.73%	99.65%	88.48%
1.00	9.2%	0%	0.48%	99.77%	87.79%

These energy-consumption overheads may seem overwhelming, but the overall overhead will be much smaller when our energy-saving mechanism is used. The time duration of DAPL operating in SLEEP and ACTIVE modes is significantly less than the time in OFF mode. According to the statistics provided by American Automobile Association Foundation for Traffic Safety [58], the average driving time for a driver is 70.2 minutes/day. Assuming that the phone moves every 5 minutes in a car and each movement lasts 5 seconds, the average energy overhead per day is 0.49%

$$\frac{OH_{active}\ \Sigma_i\ |\mathcal{T}_{active,i}| + OH_{sleep}\ \Sigma_j\ |\mathcal{T}_{sleep,j}|}{|\mathcal{T}_{day}|} = 0.49\%,$$

where $\mathcal{T}_{active,i}$ ($\mathcal{T}_{sleep,j}$) represents the time segments in ACTIVE (SLEEP) mode and $\mathcal{T}_{day}$ equals 24 hours. DAPL achieves 91.71% accuracy for real-time in-car location estimation with only 0.49% energy overhead per day (a total of 5.49% if entering car detection [20] is included)!

Let us consider the practical scenario when false positive and false negative of entering ACTIVE mode can occur. We evaluate the effectiveness of the proposed energy-saving mechanism by evaluating the tradeoff between energy consumption and localization accuracy with different parameter settings for entering ACTIVE mode. We define the threshold ratio R_{act} to be the ratio of the threshold for entering ACTIVE mode to that used for motion detection: $R_{act} = TH^*_{gyro}/TH_{gyro}$. Table 6 shows the evaluation results of the proposed energy-saving mechanism. We present FPR_e, FNR_e, energy overhead, and accuracies when R_{act} is set to 0.5, 0.75 and 1, respectively, by adjusting TH^*_{gyro}. The energy overhead is calculated based on the same settings as we calculated the ideal case while taking FPR_e into consideration. As shown in Table 6, if we set R_{act} to be a small value (0.5), DAPL will have high FPR_e (23.3%) and low FNR_e (2.7%). That is, DAPL will enter ACTIVE mode more frequently with higher energy consumption and less miss detection time. So, DAPL is able to achieve higher accuracy

(91.71%). When R_{act} increases, DAPL will have less FPR_e, but more FNR_e and less localization accuracy. This tradeoff of energy consumption and localization accuracy is a design choice. Developers can choose their preference when implementing the system. We recommend setting R_{act} to lower than 0.75 to ensure 88+% accuracy. The proposed energy-saving mechanism is shown to reduce 98.68% energy overhead (from 207.02% to only 2.73%) per day in a practical usage scenario. This is only a 7.73% overhead (compared to phone standing by) per day even when the detection of entering car is included and it is equivalent to only less than 2.5% (or 17min) reduction of actual phone usage/screen-on time per day.

6 Conclusion

We have proposed a new in-vehicle phone localization scheme, called DAPL, for mitigation of smartphone-distracted driving. DAPL is designed to operate on commodity hardware of smartphones and cars, and it does not require any additional dedicated devices to be installed in a car, thus facilitating its deployment. DAPL exploits distinct movement features to estimate the phone's location inside a car. It captures movements by built-in sensors and extracts their trends by applying coordinate transformation, movement-time offset compensation, and feature extraction. Finally, it matches the trends to a specific movement for location estimation. DAPL is shown to be able to achieve 91.71% accuracy at low energy overhead ($<2.5\%$ reduction of usage/screen-on time) in practical daily use, making it very attractive to end-users and carmakers.

References

[1] M. Richtel, "In study, texting lifts crash risk by large margin," Accessed: Mar. 06, 2017, 2009. [Online]. Available: http://www.nytimes.com/2009/07/28/technology/28texting.html?mcubz=0

[2] NHTSA, "Distracted driving: Facts and statistics," Accessed: Mar. 05, 2017, 2016. [Online]. Available: https://www.distraction.gov/stats-research-laws/facts-and-statistics.html

[3] Texas Department of Transportation, "Texas motor vehicle crash statistics," Accessed: Mar. 07, 2017, 2015. [Online]. Available: http://www.txdot.gov/government/enforcement/annual-summary.html

[4] L. Li, R. A. Shults, R. R. Andridge, M. A. Yellman, H. Xiang, and M. Zhu, "Texting/emailing while driving among high school students in 35 states, United States, 2015," *J. Adolescent Health*, vol. 63, pp. 701–708, Dec. 2018.

[5] C. Bi, J. Huang, G. Xing, L. Jiang, X. Liu, and M. Chen, "SafeWatch: A wearable hand motion tracking system for improving driving safety," in *Proc. 2nd Int. Conf. Internet-of-Things Des. Implementation*, 2017, pp. 223–232.

[6] L. Jiang, X. Lin, X. Liu, C. Bi, and G. Xing, "SafeDrive: Detecting distracted driving behaviors using wrist-worn devices," *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 1, Jan. 2018, Art. no. 144.

[7] B. Goel, A. K. Dey, P. Bharti, K. B. Ahmed, and S. Chellappan, "Detecting distracted driving using a wrist-worn wearable," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, 2018, pp. 233–238.

[8] L. Liu *et al.*, "Toward detection of unsafe driving with wearables," in *Proc. Workshop Wearable Syst. Appl.*, 2015, pp. 27–32.

[9] S. Deshmukh and O. Dehzangi, "Identification of real-time driver distraction using optimal subBand detection powered by Wavelet Packet Transform," in *Proc. IEEE 14th Int. Conf. Wearable Implantable Body Sensor Netw.*, 2017, pp. 9–12.

[10] L. Alam and M. M. Hoque, "Real-time distraction detection based on driver's visual features," in *Proc. Int. Conf. Electr. Comput. Commun. Eng.*, 2019, pp. 1–6.

[11] T. Liu, Y. Yang, G.-B. Huang, Y. K. Yeo, and Z. Lin, "Driver distraction detection using semi-supervised machine learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1108–1120, Apr. 2016.

[12] D. Tran, H. Manh Do, W. Sheng, H. Bai, and G. Chowdhary, "Real-time detection of distracted driving based on deep learning," *IET Intell. Transport Syst.*, vol. 12, pp. 1210–1219, Dec. 2018.

[13] C. Streiffer, R. Raghavendra, T. Benson, and M. Srivatsa, "Darnet: A deep learning solution for distracted driving detection," in *Proc. 18th ACM/IFIP/USENIX Middleware Conf. Industrial Track*, 2017, pp. 22–28.

[14] F. Mizoguchi, H. Nishiyama, and H. Iwasaki, "A new approach to detecting distracted car drivers using eye-movement data," in *Proc. IEEE 13th Int. Conf. Cogn. Informat. Cogn. Comput.*, 2014, pp. 266–272.

[15] M. Raja, V. Ghaderi, and S. Sigg, "Detecting driver's distracted behaviour from Wi-Fi," in *Proc. IEEE 87th Veh. Technol. Conf.*, 2018, pp. 1–5.

[16] M. L. Watkins, I. A. Amaya, P. E. Keller, M. A. Hughes, and E. D. Beck, "Autonomous detection of distracted driving by cell phone," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst.*, 2011, pp. 1960–1965.

[17] S. Im, C. Lee, S. Yang, J. Kim, and B. You, "Driver distraction detection by in-vehicle signal processing," in *Proc. IEEE Symp. Comput. Intell. Veh. Transp. Syst.*, 2014, pp. 64–68.

[18] C. Bo, X. Jian, X.-Y. Li, X. Mao, Y. Wang, and F. Li, "You're driving and texting: Detecting drivers using personal smart phones by leveraging inertial sensors," in *Proc. 19th Annu. Int. Conf. Mobile Comput. Netw.*, 2013, pp. 199–202.

[19] H. Chu, V. Raman, J. Shen, A. Kansal, V. Bahl, and R. R. Choudhury, "I am a smartphone and I know my user is driving," in *Proc. 6th Int. Conf. Commun. Syst. Netw.*, 2014, pp. 1–8.

[20] H. Park, D. Ahn, T. Park, and K. G. Shin, "Automatic identification of driver's smartphone exploiting common vehicle-riding actions," *IEEE Trans. Mobile Comput.*, vol. 17, no. 2, pp. 265–278, Feb. 2018.

[21] G. Johnson and R. Rajamani, "Smartphone localization inside a moving car for prevention of distracted driving," *Veh. Syst. Dyn.*, vol. 58, no. 2, pp. 290–306, 2020, doi: 10.1080/00423114.2019.1578889.

[22] P.-F. Ho and J.-C. Chen, "BTGuard - Using bluetooth to sense driver phone use," in *Proc. 1st ACM Int. Workshop Smart Auton. Connected Veh. Syst. Services*, 2016, pp. 68–69.

[23] Y. Wang, J. Yang, H. Liu, Y. Chen, M. Gruteser, and R. P. Martin, "Sensing vehicle dynamics for determining driver phone use," in *Proc. 11th Annu. Int. Conf. Mobile Syst. Appl. Services*, 2013, pp. 41–54.

[24] J. Wahlstrom, I. Skog, P. Handel, and A. Nehorai, "IMU-based smartphone-to-vehicle positioning," *IEEE Trans. Intell. Veh.*, vol. 1, no. 2, pp. 139–147, Jun. 2016.

[25] E. J. Wang, J. Garrison, E. Whitmire, M. Goel, S. Patel, and P. G. Allen, "Carpacio: Repurposing capacitive sensors to distinguish driver and passenger touches on in-vehicle screens," in *Proc. 30th Annu. ACM Symp. User Interface Softw. Technol.*, 2017, pp. 49–55.

[26] J. Yang *et al.*, "Detecting driver phone use leveraging car speakers," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Netw.*, 2011, pp. 97–108.

[27] M. Redmayne, "Where's your phone? A survey of where women aged 15–40 carry their smartphone and related risk perception: A survey and pilot study," *PLoS One*, vol. 12, 2017, Art. no. e0167996.

[28] Y. Cui, J. Chipchase, and F. Ichikawa, "A cross culture study on phone carrying and physical personalization," in *Proc. 2nd Int. Conf. Usability Internationalization*, 2007, pp. 483–492.

[29] Surviving cost-cutting pressures from OEMs. Accessed: Mar. 05, 2020. [Online]. Available: https://www.thefabricator.com/stampingjournal/article/shopmanagement/surviving-cost-cutting-pressures-from-oems

[30] Beyond cost reduction | The boston consulting group. Accessed: Mar. 05, 2020. [Online]. Available: https://www.bcg.com/documents/file14316.pdf

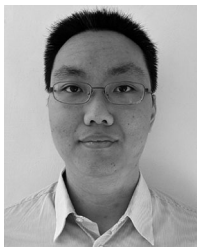
[31] Car price increases | Bureau of labor statistics. Accessed: Mar. 05, 2020. [Online]. Available: https://www.bls.gov/opub/ted/2017/mobile/prices-for-new-cars-up-5-percent-insurance-50-percent-over-last-10-years.htm

[32] How to use do not disturb while driving - Apple support. Accessed: Mar. 19, 2020. [Online]. Available: https://support.apple.com/en-us/HT208090

[33] Have driving mode turn on automatically on your Pixel phone - Pixel phone help. Accessed: Mar. 19, 2020. [Online]. Available: https://support.google.com/pixelphone/answer/9140827?hl=en

[34] Bluetooth SIG, "Bluetooth in automotives," Accessed: Oct. 12, 2018, 2018. [Online]. Available: https://www.bluetooth.com/markets/automotive

[35] NHTSA, "Distracted driving | NHTSA," Accessed: Dec. 20, 2020. [Online]. Available: https://www.nhtsa.gov/risky-driving/distracted-driving
[36] H. Li, H. Wang, L. Liu, and M. Gruteser, "Automatic unusual driving event identification for dependable self-driving," in *Proc. 16th Conf. Embedded Netw. Sensor Syst.*, 2018, pp. 15–27.
[37] B. Qi, P. Liu, T. Ji, W. Zhao, and S. Banerjee, "DrivAid: Augmenting driving analytics with multi-modal information," in *Proc. IEEE Veh. Netw. Conf.*, 2019, pp. 1–8.
[38] AT&T, "Get the AT&T DriveMode app - Wireless customer support," Accessed: Apr. 27, 2021. [Online]. Available: https://www.att.com/support/article/wireless/KM1000730/
[39] B. Molen, "Sprint launches drive first Android app to curb texting and driving, keep chatty teens at bay | Engadget," Accessed: Apr. 27, 2021. [Online]. Available: https://www.engadget.com/2011–09-12-sprint-launches-drive-first-android-app-to-curb-texting-and-driv.html
[40] No radiation for you, "EMF in cars," Accessed: Aug. 05, 2017, 2010. [Online]. Available: http://www.norad4u.com/knowledge/emr-sources-by-location/emf-in-cars
[41] R. Hareuveny *et al.*, "Characterization of extremely low frequency magnetic fields from diesel, gasoline and hybrid cars under controlled conditions," *Int. J. Environ. Res. Public Health*, vol. 12, pp. 1651–1666, 2015.
[42] D. Chen, K.-T. Cho, S. Han, Z. Jin, and K. G. Shin, "Invisible sensing of vehicle steering with smartphones," in *Proc. 13th Annu. Int. Conf. Mobile Syst. Appl. Services*, 2015, pp. 1–13.
[43] P. Zhou, M. Li, and G. Shen, "Use it free: Instantly knowing your phone attitude," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 605–616.
[44] U.S. Department of Transportation, "Safety pilot open dataset," Accessed: Dec. 14, 2021. [Online]. Available: https://datahub.transportation.gov/Automobiles/Safety-Pilot-Model-Deployment-Data/a7qq-9vfe
[45] C. M. Bishop, *Pattern Recognition and Machine Learning*. Berlin, Germany: Springer, 2006.
[46] Hyundai bluetooth pairing instructions. Accessed: Aug. 18, 2019. [Online]. Available: https://www.hyundaibluetooth.com/content/Bluetooth/Hyundai/us/en/PairingInstructions/InstructionSteps.html?vYear=2018&vModel=Elantra&vAudio=Display Audio&vAudioId=62&vAudioType=16&pCarrier=T-Mobile&pManu=Apple&pModel=iPhone 6s&pVersion=iOS 9.3.2&pPa
[47] Tesla Model S X Bluetooth Setup - YouTube. Accessed: Aug. 18, 2019. [Online]. Available: https://www.youtube.com/watch?v=6EyYyuDmo_4
[48] Jaguar XF Bluetooth Phone Pairing - YouTube. Accessed: Aug. 18, 2019. [Online]. Available: https://www.youtube.com/watch?v=a1dsmocus9I
[49] How to pair your Bluetooth phone to you car- Uconnect Walkthrough for Chrysler, Dodge or Jeep - YouTube. Accessed: Aug. 18, 2019. [Online]. Available: https://www.youtube.com/watch?time_continue=30&v=1teZTDzbL6k
[50] Step-by-step chevrolet bluetooth setup guide. Accessed: Aug. 18, 2019. [Online]. Available: https://www.autonationchevroletnorth.com/Chevrolet-Bluetooth-Setup
[51] How to pair your phone with SYNC | Official Ford Owner Site. Accessed: Aug. 18, 2019. [Online]. Available: https://owner.ford.com/support/how-tos/sync/sync/setup/how-to-connect-or-pair-my-phone-with-sync.html
[52] A. Vassilev, A. Ferber, C. Wehrmann, O. Pinaud, M. Schilling, and A. R. Ruddle, "Magnetic field exposure assessment in electric vehicles," *IEEE Trans. Electromagn. Compat.*, vol. 57, no. 1, pp. 35–43, Feb. 2015.
[53] N. G. Ptitsyna, A. Ponzetto, Y. A. Kopytenko, V. S. Ismagilov, and A. G. Korobeinikov, "Electric vehicle magnetic fields and their biological relevance," *J. Sci. Res. Reports*, vol. 3, pp. 1753–1770, 2014.
[54] C. C. Chan, "The state of the art of electric, hybrid, and fuel cell vehicles," *Proc. IEEE*, vol. 95, no. 4, pp. 704–718, Apr. 2007.
[55] American Association of State Highway and Transportation Officials, *A Policy on Geometric Design of Highways and Streets*, American Association of State Highway and Transportation Officials, 2018. [Online]. Available: https://store.transportation.org/Item/CollectionDetail?ID=180
[56] The Babcock Law Firm LLC, "4 most common places where car accidents happen," Accessed: Mar. 08, 2019. [Online]. Available: https://www.injurylawcolorado.com/legal-news/auto-accidents/common-car-accident-locations.htm
[57] NHTSA, "Traffic safety facts annual report tables," Accessed: Mar. 08, 2019. [Online]. Available: https://cdan.nhtsa.gov/stsi.htm
[58] American Automobile Association Foundation for Traffic Safety, "American driving survey 2014–2015," Accessed: Mar. 05, 2017, 2016. [Online]. Available: https://www.aaafoundation.org/american-driving-survey-2014%E2%80%932015-1



Chun-Yu Chen received the BS degree in electrical engineering and the MS degree in communication engineering from National Taiwan University, Taiwan, in 2012 and 2014, respectively. He is currently working toward the PhD degree in computer science and engineering at the University of Michigan, Ann Arbor, Michigan, advised by Prof. Kang G. Shin. His research interests include vehicle cyber-security/safety and mobile computing/privacy.

Kang G. Shin (Life Fellow, IEEE) is currently the Kevin & Nancy O'Connor professor of computer science with the Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, Michigan. His current research focuses on QoS sensitive computing and networking as well as on embedded real-time and cyber-physical systems. He has supervised the completion of 88 PhDs, and authored/coauthored about 1,000 technical articles, a textbook and more than 60 patents or invention disclosures, and received numerous best paper awards, including 2019 Caspar Bowden Award for Outstanding Research in Privacy Enhancing Technologies, the best paper awards from the 2011 ACM International Conference on Mobile Computing and Networking (MobiCom'11), the 2011 IEEE International Conference on Autonomic Computing, the 2010 and 2000 USENIX Annual Technical Conferences, as well as the 2003 IEEE Communications Society William R. Bennett Prize Paper Award and the 1987 Outstanding IEEE Transactions of Automatic Control Paper Award.