In-Vehicle Driver Detection

Using Mobile Phone Sensors

Jeffrey (Shih-kai) Shen

Advisor: Romit Roy Choudhury

SUBMITTED FOR GRADUATION WITH DEPARTMENTAL DISTINCTION IN ELECTRICAL AND COMPUTER ENGINEERING

Accepted for ACM MobiSys June 2011 Poster

Acknowledgements

I would first like to thank my adviser Professor Romit Roy Choudhury for his guidance and mentorship for the past one and half years. I would also like to thank my partners Hon Chu and Vijay Raman for putting the time and effort into the project to make it a success. Lastly I would like to thank Aman Kansal for his insights and support throughout the whole process. Furthermore, I would like to thank everyone in the SynRG group for their help and for a memorable senior year.

Abstract

In this research project, I present a mobile phone based sensing system that determines if a mobile user in an automobile is the driver or a passenger. This system will help the mobile device aware of its user's attention span and appropriately control the delivery of notifications such incoming calls and context based reminders generated with changing user location as to limit distractions during a drive. The in-vehicle driver detection system uses only sensors and resources available on the mobile phone. It does not rely on any additional custom sensor deployment, wearable devices, or any communication with in-vehicle sensors or electronics. Furthermore, the user can carry the phone in different pockets or in different orientations. In the paper, I will describe the various sensing mechanisms, features, and designs to address the various challenges encountered in the project.

Introduction

A. Related Works

The vast array of sensors on mobile phones enables sensing algorithms to infer information about users' activities. Prior works have developed algorithms to detect various interesting user activities. Specifically, several research papers have addressed similar research problems related to this research project. The most pertinent paper uses user activity sensing to determine the modes of transportation the user is in [1,2]. Furthermore, other studies have done extensive work to address challenges to measure traffic and travel times [3], automatically classify events in people's lives [4], map urban environmental characteristics [5], and sense health related behaviors [6]. My colleagues and I will add to and extend this rich repertoire of user activity sensing.

B. Motivation and Project Goals

People spend a significant of time in a car each day. In the U.S., the average is 86 minutes/day, and in Europe, the average is 43 minutes/day [7]. Driving safety has become an increasing concern as mobile phone usage increased over the past decade. Many driving accidents are caused by driver being distracted due to mobile phone functions such as calling or texting [8, 9]. The motivation of the project is to provide the system that can identify the user as the driver or passenger to eliminate these distractions. For example, if the mobile phone user is the driver, the phone can divert distractions by suppressing in-call and texting functions as to avert hazardous situations. While, if the phone detects that the user is the passenger, the passenger will have ample time to respond to emails, make calls or even text. Thus, those phone functions will not be disabled. Attention based notification delivery is crucial to not only user experience [10,11] but also driving safety. The system, hereafter known as Driver Detection System (DDS), in making mobile device aware of the context of the user's attention while driving is a positive benefit for the research community as well as the public.

C. Research Challenges

There are several research challenges that are encountered and addressed in this research project. First, the main challenge in detecting the user's context is that the phone should not be worn or placed in a specific part of the user's body. This means that the system should be robust if the phone is placed or worn in many different circumstances. For example, the phone can be in the lower or upper body or even carried in a handbag. It is easy to see that there are many cases to be considered and pose as non-trivial challenges.

Second, many of the signatures that identify the user as the passenger or driver are short-lived events on the order of seconds. The challenge is that these features must be captured and appropriately interpreted. Furthermore, the result must be quickly processed and interpreted so the phone can quickly return the user's context as the driver or the passenger. Therefore, this does not allow the system to infer these behaviors offline from collected data. In addition, the system is implemented on a mobile system where energy is limited. Therefore, this implies that the system must be efficient in its sensing and data processing. Excessive communication or continuous sensing might not be feasible for DDS.

Finally, I am presenting a system that does not use any additional hardware or does not require additional communication with any in-vehicle sensors or electronics. Also, large sensors cannot be added to the mobile device. Thus, this requires the system to only use the sensors and resources available to the mobile phone.

I am presenting a mobile phone sensing system that determines the user in an automobile as the driver or a passenger using only sensors available on the phone. In the sections below, I will discuss the various user micro-movements that can be detected by the system to allow the phone to identify the user's context. The detection will be broken down into several distinct detection algorithms: front vs. back, driver side vs. passenger side, and other signatures. These algorithm performances and accuracies will be discussed. Furthermore, I will also discuss energy consumption and energy constraints associated with using the system. The system is evaluated on iPhone OS 4.0 and Android OS 2.3 devices used by multiple users.

D. Existing Solutions

The discussion will not be completed without some discussion of existing solutions to eliminate driver distraction or suggested solutions to identify the driver in the car. First, there are existing software applications that lock out the phone when the phone is detected to be travelling above 30 miles/hour. The major problem for these applications is that the phone cannot differentiate between the driver and the passenger. Furthermore, the application cannot differentiate between different modes of transportation such as bus or train and will lock the user out of the phone's functionalities. Thus, these applications are limited in context awareness.

Several alternatives to DDS to identify the phone's user as the driver currently are feasible. The phone can have a near field communication (NFC) radio. The car can have a NFC radio for key-less entry. If the two systems can communicate with each other and the car's system informs the phone which door is used, the phone can determine if the user is the driver. This approach is feasible; however, even among new cars, NFC radios are installed only in the small fraction of the cars and only on the front doors. This suggests that the approach will not work for rear passengers and for cars without the key-less entry features.

The second alternative is to use audio to detect the phone's position in the car. However, no feasible model in existing literature exists yet to provide a robust form of driver detection system. The third alternative is to utilize the pressure sensors that are placed in car seats. These sensors are used to activate the airbags near the seats. If the only driver seat weight sensor is activated, then the phone clearly belongs to the driver. However, if there are more than one seats activated in a car, a mobile phone has no means to know which seat the phone is on.

Therefore, it is much easier to deploy software solutions that use only the existing sensors on the phone than to employ hardware solutions. The software solution is much practical and easier to distribute.

Unlike hardware modifications require significant customization that would enable the phone to interact with the car's electronics. My colleagues and I consider that the system will only have the phone and built-in sensors are available for the solution.

E. My Contributions

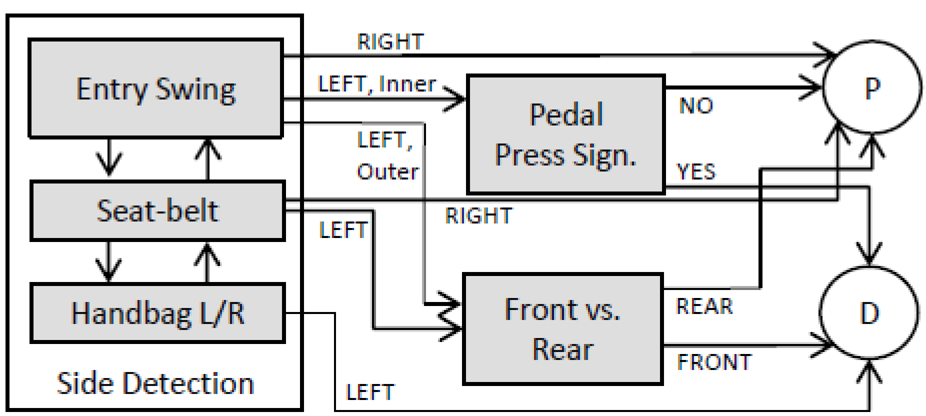
As noted above, several people worked on the project equally together on this research project. In this research paper, I will specifically denote and elaborate the pieces of the system that I worked on. These include the Brake SVM, Seatbelt SVM, Front vs. Back detection, and implementing the whole DDS on the Android OS 2.3. In addition, I will also introduce and summarize the pieces of system that others worked on as to put my work in context of the whole system. This will, one, enable readers to understand the whole system as a whole. Also, it will enable readers to understand how the whole system is integrated, implemented, and evaluated.

Solution Design

Driver Detection System

The key intuition behind our approach in DDS is to use the following phone sensors such as the accelerometer to measure acceleration along 3 axes and the gyroscope to measure the rotation along 3 axes, denoted by roll, pitch, and yaw. These sensors will provide readings that will be used to detect user movement patterns in the automobile to find the user's context as the driver or the passenger.

Figure 1: DDS Decision Flow. D = driver, P = passenger



In this section, I will describe the various components that will make up the DDS system. As seen above in Figure 1, the DDS Decision Flow is made out of five distinct components in grey. They are as of the following: Entry Swing, Seat-belt, Handbag L/R, Pedal Press Signature, and Front vs. Rear. The five components are utilized in unison to determine the user is the driver or the passenger. As seen above, the Entry Swing, Seat-belt, and Handbag L/R are responsible for the Side Detection. The Pedal Press Signature is an additional signature to aid to determine if the phone belongs to the driver. Lastly, Front vs. Rear is used to determine if the phone is in the front of the car or in the rear of the car to determine the appropriate context of the user.

The **Entry Swing** block determines if the user enters from the right or left side of the automobile. This signature is captured when the phone is in the user's lower body. The intuition behind this signature is that the movement of entering the car can be separated into two significant parts. There are two distinct signatures when the phone is on the leg closest to the car door (inner foot) and or when the phone is on the leg furthest from the car door (outer foot). The accelerometer, compass, and gyroscope data are used in capturing the entry swing signature.

The **Seat-belt** signature is used to detect right or left entry into the car when the phone is on the upper body. This signature is specifically used to detect when the user puts on the seatbelt. For example, the user on the right side of the car will wear his/her seatbelt by turning to the right to get the seatbelt and then turning to the right to fasten the seatbelt. This same motion is exactly the reverse for the person on the left side of the car. Accelerometer, compass, and gyroscope data are used to capture this distinct signature.

The **Handbag L/R** block determines left vs. right for a phone carried in a handbag. This is activated when the other two side detection signatures fail. The algorithm is similar to the Front vs. Rear detection described below.

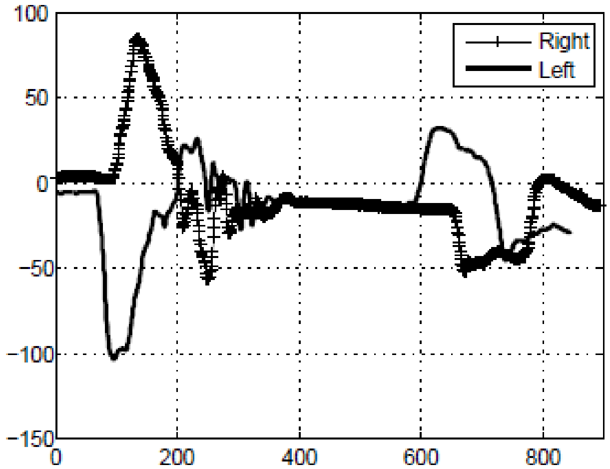
Front vs. Rear detection block determines if the user's phone is in the front or the rear of the car. This block is only required if other branches shown in Figure 1 is inconclusive. The block uses communication with a cloud service with the phone's audio data to determine if the user is in the front or the rear. The advantage in this model is that no direct communication is necessary among the phones in the car so the user does not need to enter a password for Bluetooth communication. In this component, microphone data from the phone is used.

Sensing Algorithms

I will individually go through the components that I have built in this project. For these components, I will describe the sensing algorithms in detail and the intuition behind these signatures.

The **Seat-belt** signature is based on the rotation of the user's body when the user is putting on the seatbelt. In the global reference frame, the rotation of the body shows up on the gyroscope's yaw axis. This distinct signature is only apparent for the phone when it is carried in the upper body pocket such as the shirt pocket. When the phone is in the lower body, there is no significant rotation when the seatbelt is worn. Sample signatures are captured on the gyroscope for both left and right entry and are shown in Figure 2.

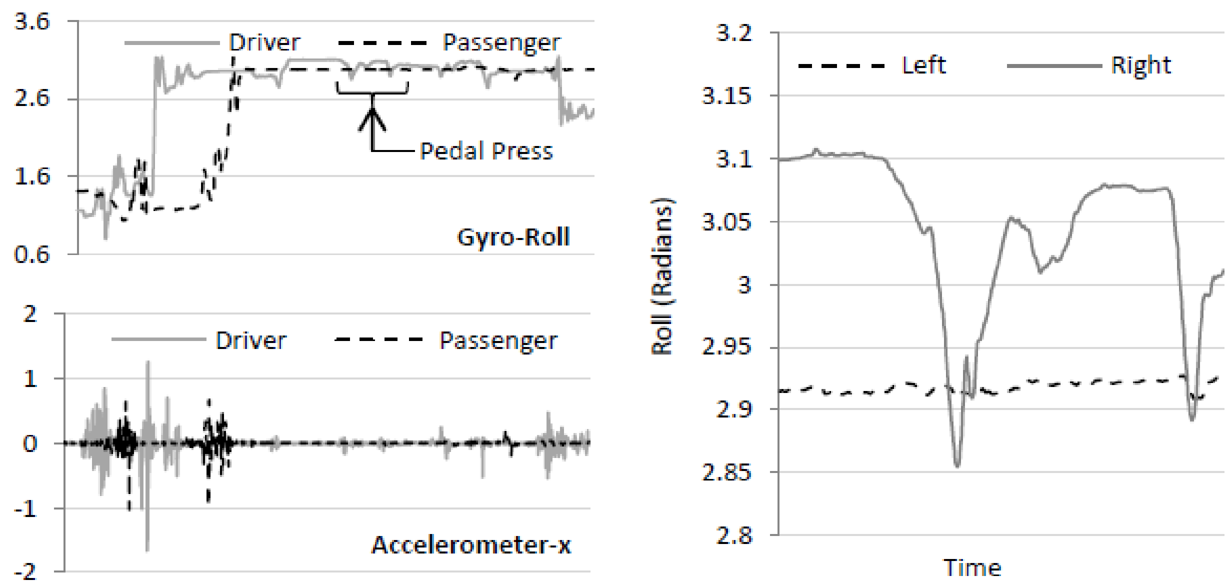
Figure 2: Seat-belt Signature



I use a machine learning classification algorithm, support vector machine (SVM) [12], to classify input signals into left entry, right entry, and no-signature cases. The SVM is trained on these following features: first derivative of the yaw axis, variance of the yaw, and the variance of the three axes of the accelerometer data. The accelerometer data is still used in the SVM because the readings help with distinguishing rotation around the yaw axis from other signals from the no-signature cases. The no-signature cases are made out random sampling of different scenarios such as walking, non-complete seatbelt signatures, and other car entry signatures.

The **Brake** signature detects the leg movement when the driver presses down on the brake or the gas pedal. The sample trace of a driver and a passenger is shown below in Figure 3. This micro-movement signature only occurs when the driver's inner leg (right leg) presses down on the pedals and when the phone is suspected to be on the inner leg. The signature consists of three distinct periods. First there is a sudden change in the pitch or the roll when the user presses on the pedal. Second, there is a period of no activity for a period when the pedal is kept pressed. Finally, there is again another change of pitch or roll when the pedal is finally released. This movement is actually more pronounced for the brake pedal than the gas pedal. For the SVM, several features are used such as second derivative of the gyroscope pitch and roll and zero crossings for the second derivative. The second derivative was found to be the most useful feature for the SVM because a pulse like pattern, corresponding to increase in pitch/roll, followed by no change in values and then a decrease a decrease of pitch/roll, fits the brake signature well.

Figure 3: Brake Signature



The **Front vs. Rear** block is used when the user is detected to be on the left side of the phone but the phone is not in the appropriate places to label the phone as the driver. In essence, this block is needed to differentiate the driver from the passenger who seats or enters from the car on the left side. There two key challenges in using audio as the detection method. First, the absolute amplitude of a turn signal sound is not useful because it can vary across cars and across different situations. Therefore, the detection will be using comparisons between the observed amplitudes of the phones present in the car. Furthermore, since Bluetooth pairing requires user intervention, I chose a back-end cloud server to process the data. The back-end server must be able to determine if two or more phones uploading data to the server are from the same vehicle based on sensor readings from the phones. This feature will need accelerometer data and timestamp to aid the groupings.

The second challenge for the audio detection method is that the environment will be very noisy. There will be many other sounds such as road noise and music, which can distort the turning signal sound. This problem is minimized by using a band-pass filter to separate the turn signal sound. Some errors can realistically occur since some frequency components in music or other audio sounds can lie within the band-pass. However, based on experimental data, the filter has worked quite well. The filter design specifications are listed below in Table 1.

Table 1: Front vs. Rear Band-pass Filter

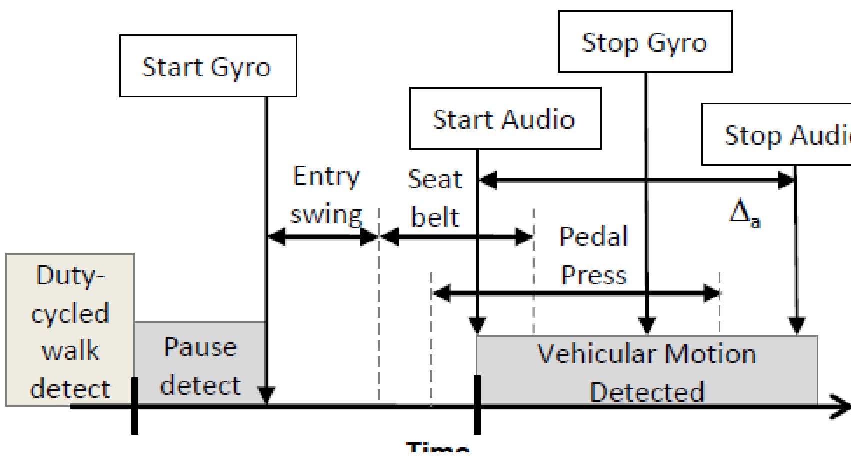
Sampling	Stop Band	Pass Band	Ripple
44.1 KHz	<2.9kHz, > 3.1kHz	2.95-3.05kHz	0.057501127785

Event Triggers

DDS needs several triggers to activate the appropriate sensing and inference because limited energy constrains the system from continuous sensing. On this, DDS uses duty-cycled approach to detect when data should be collected and then uses sensor readings to turn sensors on or off when sufficient data has been collected. The timeline for sensor activations on trigger events is shown in Figure 4.

The detection of walking using the accelerometer is a well studied sensor event and in DDS, the system uses a simple amplitude based method to detect walking. The detection structure is duty cycled – the sensor is activated for every one second for every minute. When walking is detected, the accelerometer is turned on continuously to detect if the user pauses. When the pause does occur, the gyroscope is turned on and the rest of the sensor activations are shown in Figure 4.

Figure 4: Timeline for Sensor Activations on Trigger Events



In addition, the start of vehicular motion is also needed in DDS. This has been well established from prior works [13] using accelerometer data. Lastly, timeout triggers are also employed to deactivate triggers to save energy. The gyroscope data collection is stopped fifteen seconds after the vehicular motion is detected. This is because from experimental experiments, the users will wear the seatbelt in most cases within that time frame.

Evaluation

Experimental Data Collection

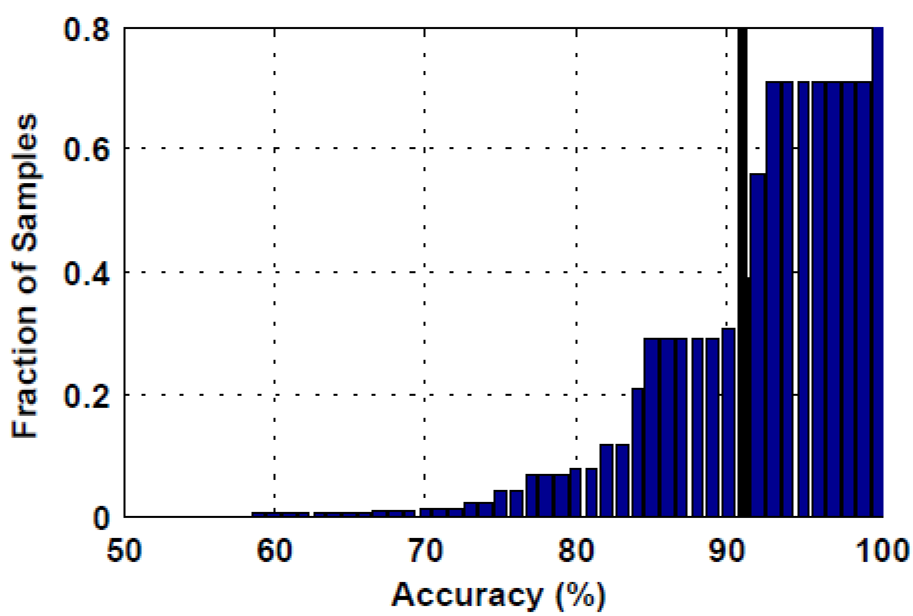
The overall system is evaluated on sensor data collected on iPhone OS 4.0 based iPods and Android OS 2.2 based Nexus Ss. The data are collected by three distinct users in realistic settings with the phone carried on different positions of the body without controlling the phone's orientation. Multiple data traces were collected from each user – leading up to 40 or more samples for each of the micro-signatures.

Accuracy

One way to evaluate the overall performance of DDS is to quantify the error for each of the sensing blocks. Furthermore, another technique is to verify if the inference methods trained on one user will allow the DDS system to work on another user. This will ultimately allow the system to be used without per user training. To this end, the results presented in this section are obtained by using 80-20 cross validation tests. This is done by using 80% of the data for training while using the remaining 20% for testing. To minimize bias in the splits, I present 100,000 different splits among the samples considered for each one test run. Below, I will present the cumulative distribution function (CDF) for the components that I worked on.

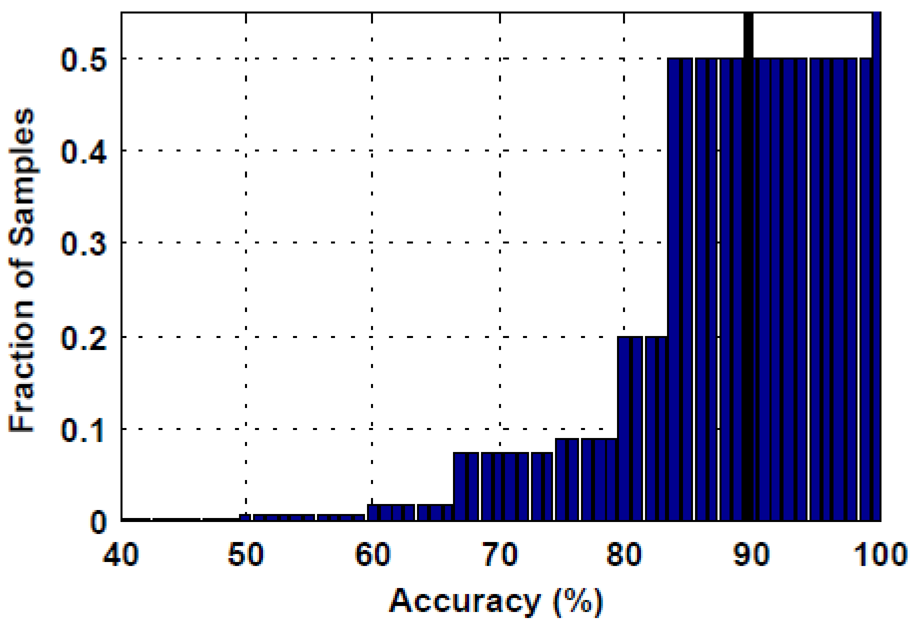
The **Seat-belt** micro-signature is evaluated with the cross validation method on all users sampled collected with the method described above. The accuracy CDF is shown below in Figure 5. The accuracy is high across all run and the mean accuracy is 91.08%, as marked by the solid black line in the figure.

Figure 5: CDF of Correct Detection Percentage on Seat-belt Signature



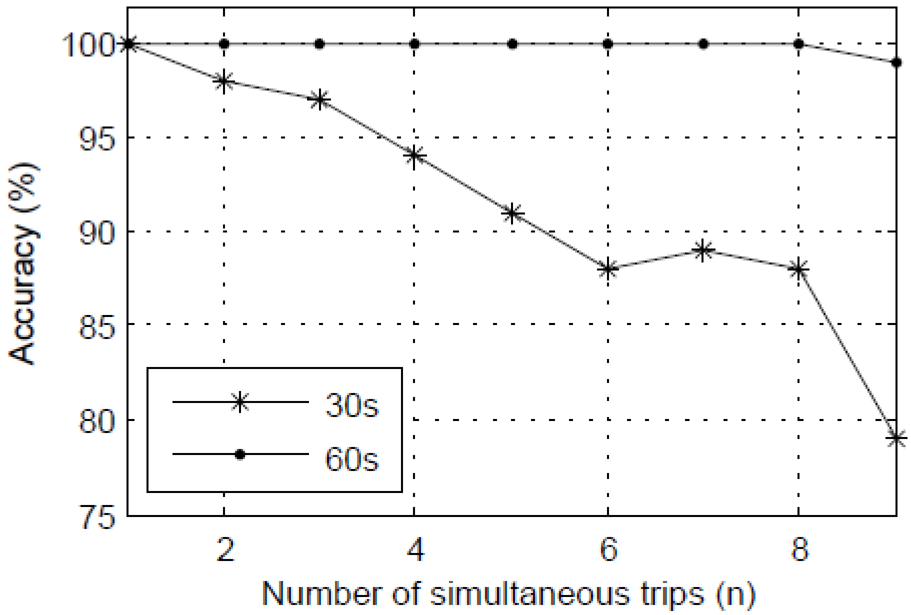
For the **Brake** signature, the same evaluation technique is also used. The CDF of correct detection percentage is shown in Figure 6. The overall accuracy for this signature is 89.78%.

Figure 6: CDF of Correct Detection Percentage on Brake Signature



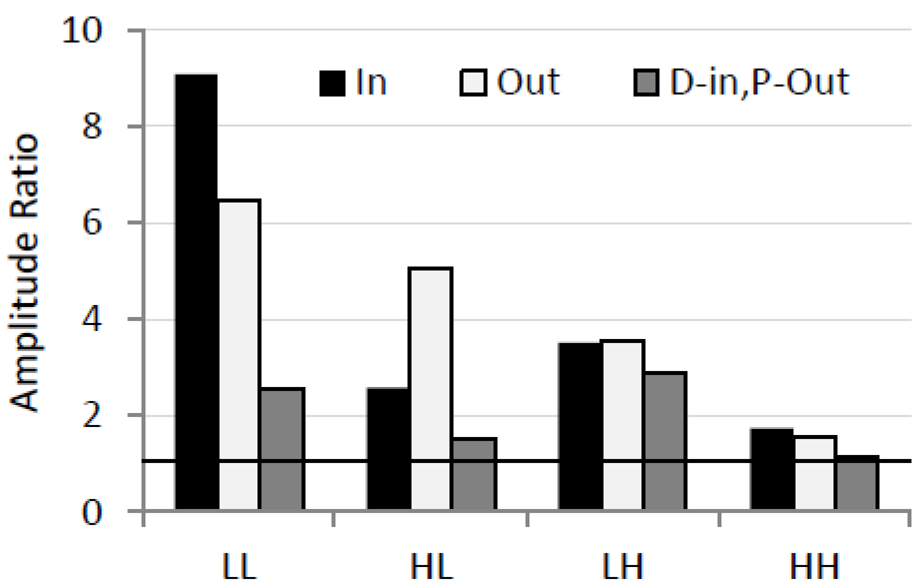
In the **Front vs. Rear** detection block, the first step in evaluating this system component, we need to see if the system can detect which cloud uploads come from the same car. This is based on matching similar location areas, time stamps, and the shape of the accelerometer data. In this research project, matching similar location areas and time stamps is trivial so we will show the results based on only accelerometer readings. Taking 100 accelerometer traces from different trips and treating each trace as an upload from a distinct phone, we divide the traces into group of up to n simultaneous trips with 2 phones within the same automobile for each trip. Accuracy is measured by the number of false positives and false negatives. In Figure 7, two different lengths of uploaded traces are shown with n varying from 2 to 10 simultaneous trips. As expected, it is apparent that the longer the trace, the more accurate the algorithm is. In the data collection, the accelerometer is sampled at 50Hz.

Figure 7: Cloud Server Matching of Uploading Data Traces from Multiple Phones



The second step of the Front vs. Back process block is to compare the filtered audio data's amplitudes between the different phones. We collected test data with both music on or off and on trips where road noise is high or low. The cases where there is High volume music and High road noise as HH. The cases where there is High volume music and Low road noise as HL. The ratio of maximum absolute magnitudes measured by the phone in the front and the phone in the back is shown In Figure 8.

Figure 8: Ratio of Audio Amplitudes for Driver and Rear Passenger in Different Scenarios



L indicates low and H indicates high. First L or H refers to music and the second to road noise

In this setup, the phone can be inside pockets or outside of pockets. The worst case scenario for the amplitude comparison is when the driver's phone is inside the pocket but h the passenger's phone is outside. The main concern in this detection block is that errors can arise from several mitigating factors. First, the driver's phone can be in a more sound absorbent material than the passenger's pocket. Therefore, we also conducted the experiments with varying degree of pocket thickness and cloth thickness. Overall, the mean accuracy across all trials was 95.83%. In the cases of High music volume and High road noise, the amplitude ratio is much lower than other trials.

The overall accuracy of the whole DDS system depends on which branch the decision flow is taken. This ultimately depends on how the phone is carried by the user. Let the probability of error for car entry, seatbelt, pedal press, handbag, and front vs. rear detection blocks to be denoted as p_{es}, p_{sb}, p_{pp}, p_{hb}, and p_{fr}. The probability of error includes all cases where the result was false positive or false negative. For example, for a driver carrying the phone in the upper body pocket, the probability of error will be 1-(1-p_{sb})(1- p_{fr}). For the passenger with the phone in a lower body right pocket the probability of error will be p_{es}. Using these probability of errors, we find that p_{es} = 0.152, p_{sb} = 0:0892, p_{pp} = 0:102, and p_{fr} = 0:041. Plugging the probability of errors into the different branches of the decision flow shown in Figure 1, we get that DDS will perform with 76.23% - 91.08% accuracy. Actual accuracy will be higher because the presence of alternate inference paths shown above.

Energy Consumption

When the phone is in the idle state, the phone consumes on average 5.89mW. In the passive mode, the application uses the accelerometer to detect walking which takes on average 275.87mW. We do duty cycle for 1 second every minute in DDS which uses a background of 4.49mW. If a phone has a battery of 1000mAh, the extra overhead is only 2.9% of the battery capacity per day.

After walking is detected by the background passive mode, the accelerometer is turned on continuously to detect a pause. The rest of the power consumption activities are outlined in Table 2 and will follow the timeline in Figure 4. Furthermore, uploading up to 12kB of data to the cloud server using HTTP connection will take 10J [14]. A quick example will illustrate how much energy is typically consumed in entering a car.

Table 2: DDS Application Activity

App. Activity	Power (mW)
A,G,C+Detection SVMs	969.09
A,G,C sensing only	419.1
Detection SVMs only	844.8
Audio sensing only	187.91
A,G,C + audio + Pedal SVM	947.63
A,G,C + audio sensing	491.18

In this example, the user walks for 5 minutes to the car. During this phase, the battery will be continuously drained with an average of 275.87mW. At the pause, in front of the car door, the gyroscope and compass are activated along with the seatbelt and car entry signature SVM detection algorithms. After the pause, the user enters the car and wears the seatbelt within 15 seconds, which starts the audio sensor (microphone). Then, the audio will run for 3 minutes to collect the turn signal sound and the audio clip is uploaded to the cloud server. In this example, the total energy used is 155J which is 1.1% of the total phone battery capacity. In the cases where the user starts walking and does not enter a car, continuous accelerometer sampling consumes 16.5J per minute.

Overall, energy consumption depends on user behavior and if a user walks for long durations 5 to 10 times a day, the battery life is reduced by 5-15%. This is a significant but justifiable decrease of battery life. However, for users who charge their batteries every night, the convenience of automatically enabling and disabling notifications when driving is definitely useful.

Conclusion

I presented DDS – a system that uses sensors available on mobile phones to identify the user as the driver or a passenger. This research project is an important step towards enabling user attention aware information delivery on mobile phones. Sensing the availability of user's attention is useful to prioritize the delivery of different notifications such as phone calls or texting.

The evaluation of DDS shows that with the specified sensors, a phone placed in an unknown orientation or worn in some unknown manner can correctly identify the context of the user as the driver or the passenger with the help of micro-signatures. Furthermore, experimental evaluations show that DDS achieves a reasonable accuracy without requiring extra training on an individual user basis. Lastly, using specific markers, DDS can lower the power consumption overhead significantly on the mobile phone while still processing computational intensive sensing algorithms.

In all, this research project adds to the rich array of user sensing algorithms introduced earlier in related works. These sensing algorithms are continually evolving and are becoming more sophisticated as the array of sensors on mobile platforms become more powerful and plentiful. These context aware techniques will only become more useful as more user states are researched and implemented. These techniques will not only manage user contexts but will also take appropriate actions on behalf of the user when feasible. In the future, sensors will only become more ubiquities and will continue to play an integral part in our daily lives.

References

1. L. Liao, D. Fox, and H. Kautz. Learning and inferring transportation routines. In Proceedings of the 19th national conference on Artificial intelligence, AAAI'04, pages 348–353. AAAI Press, 2004.

2. A. Thiagarajan, J. Biagioni, T. Gerlich, and J. Eriksson. Cooperative transit tracking using smart-phones. In ACM SenSys, 2010.

3. A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones. In Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, SenSys '09, pages 85–98, New York, NY, USA, 2009. ACM.

4. E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the application. In Proceedings of the 6th ACM conference on Embedded network sensor systems, SenSys '08, pages 337–350, New York, NY, USA, 2008. ACM.

5. R. K. Rana, C. T. Chou, S. S. Kanhere, N. Bulusu, and W. Hu. Ear-phone: an end-to-end participatory urban noise mapping system. In Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN '10, pages 105–116, New York, NY, USA, 2010. ACM.

6. S. Consolvo, D. W. McDonald, T. Toscos, M. Y. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, I. Smith, and J. A. Landay. Activity sensing in the wild: a field trial of ubifit garden. In Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems, CHI '08, pages 1797–1806, New York, NY, USA, 2008. ACM.

7. F. Alt, D. Kern, F. Schulte, B. Pfleging, A. Sahami, and A. Schmidt. Enabling micro-entertainment in vehicles based on context information. In 2nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI), November 2010.

8. I.-M. Jonsson and F. Chen. In-vehicle information system used in complex and low traffic situations: impact on driving performance and attitude. In Proceedings of the 4th international conference on Universal access in human-computer interaction: ambient interaction, UAHCI'07, pages 421–430, Berlin, Heidelberg, 2007. Springer-Verlag.

9 D. Hibberd, S. Jamson, and O. Carsten. Managing in-vehicle distractions - evidence from the psychological refractory period paradigm. In 2nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutomotiveUI), November 2010.

10 J. Gluck, A. Bunt, and J. McGrenere. Matching attentional draw with utility in interruption. In Proceedings of the SIGCHI conference on Human factors in computing systems, CHI '07, pages 41–50, New York, NY, USA, 2007. ACM.

11 D. Billsus, D. M. Hilbert, and D. Maynes-Aminzade. Improving proactive information systems. In Proceedings of the 10th international conference on Intelligent user interfaces, IUI ’05, pages 159–166, New York, NY, USA, 2005. ACM.

12 T. Choudhury, S. Consolvo, B. Harrison, J. Hightower, A. LaMarca, L. LeGrand, A. Rahimi, A. Rea, G. Borriello, B. Hemingway, P. Klasnja, K. Koscher, J. A. Landay, J. Lester, D. Wyatt, and D. Haehnel. The mobile sensing platform: An embedded system for capturing and recognizing human activities. IEEE Pervasive Computing Magazine, 2008.

13 E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In Proceedings of the 6th ACM conference on Embedded network sensor systems, SenSys ’08, pages 337–350, New York, NY, USA, 2008. ACM.

14 D. Lymberopoulos, A. Bamis, T. Teixeira, and A. Savvides. Behaviorscope: Real-time remote human monitoring using sensor networks. In Proceedings of the 7th international conference on Information processing in sensor networks, IPSN ’08, pages 533–534, Washington, DC, USA, 2008. IEEE Computer Society.