Driver Classification Based on Driving Behaviors

Cheng Zhang[1], **Mitesh Patel**[2], **Senaka Buthpitiya**[2], **Kent Lyons**[3], **Beverly Harrison**[4]
Gregory D. Abowd[1]
Georgia Institute of Technology[1], Yahoo Labs[2], Technicolor Research[3], Microsoft[4]
chengzhang@gatech.edu, miteshp@yahoo-inc.com, senaka@yahoo-inc.com,
kent.lyons@technicolor.com, 00beverly@gmail.com, abowd@gatech.edu

ABSTRACT
In this paper we develop a model capable of classifying drivers from their driving behaviors sensed by only low level sensors. The sensing platform consists of data available from the diagnostic outlet (OBD) of the car and smartphone sensors. We develop a window based support vector machine model to classify drivers. We test our model with two datasets collected under both controlled and naturalistic conditions. Furthermore, we evaluate the model using each sensor source (car and phone) independently and combining both the sensors. The average classification accuracies attained with data collected from three different cars shared between couples in a naturalistic environment were 75.83%, 85.83% and 86.67% using only phone sensors, only cars sensors and combined car and phone sensors respectively.

Author Keywords
Driver Identification; Driver behavior modeling; Sensor fusion; Smartphones;

ACM Classification Keywords
H.4 Information System Applications : Miscellaneous

INTRODUCTION
Driving is an integral part of everyday life for many people and we are seeing interesting technologies integrated with cars including navigation systems, adaptive cruise control (ACC) and lane-keeping assist systems (LKAS). Many of these systems utilize sensing mechanisms to monitor the car, the environment and the driver. As such, the car is an interesting platform for exploring new ubiquitous computing applications. Unfortunately, these car sensing and computing platforms tend to be proprietary and closed. It is very difficult for developers or researchers to leverage these platforms to build applications.

There are standards for getting some of the car sensing data. Specifically, the On-Board Diagnostic (OBD-II)[1] protocol provides information about the car that is traditionally used to perform car maintenance tasks. Mobile phones offer another platform for rich sensing that can be used to infer information about the car and its driver. As users upgrade their phones, the old phone, which is often a very capable sensing platform, might be put in the car and serve a useful role. Alternatively, these mobile platforms might be integrated into the car as possible open platforms (e.g. Android Auto).

In this project, we explore identifying the driver of the car using the sensing capabilities available on these platforms. There are several applications where such a driver classification could be beneficial. For example, it would be useful for the car and user's mobile phone to streamline the in-car mobile phone experience. Using the visual and touch interface of a phone while driving is dangerous and in many locations illegal. Shifting to a hand-free, eyes-free interface is a possibility but requires extra steps for the user to enter that mode. In contrast, using the traditional phone interface while a passenger in the car poses no problems. Determining who is driving the car (versus a passenger) and communicating that information to the mobile devices in the car could help automatically set the appropriate modes or offer other abilities to personalize the in-car experience. A phone might have a unique identified owner and off-the-shelf techniques can classify if the phone is in a moving vehicle, however, just knowing that a phone (and its owner) is in the car is not sufficient information to adapt a user interface when there are multiple people in the car. The person could be a driver or a passenger.

In this paper, we develop driver identification models using the sensors available on smartphones along with the limited car sensor information available through the OBD-II protocol. We explore using each of these sensing platforms independently and the possibility of fusing the data sources together. We present three experiments from two different data sets. The primary aim of these experiments is to test the capability and the robustness of the classification model. Our first data set involves 14 drivers each driving three different cars and is designed to understand the machine learning challenges in this domain. Our second data set is more naturalistic whereby we collect everyday data from pairs of people that share a car. Our experiments detail the tradeoffs between the number of unique driver classifications we need to make, the impact of different sensor data on classification, and the differences between the more controlled versus naturalistic driving data sets.

[1] http://en.wikipedia.org/wiki/OBD-II_PIDs

IUI'16, March 07-10, 2016, Sonoma, CA, USA.

DOI: http://dx.doi.org/10.1145/2856767.2856806

RELATED WORK

Researchers have developed a number of solutions for understanding different aspects of user behavior in cars. These systems often use state of the art algorithms in computer vision, low level sensor data and fusion techniques and machine learning algorithms.

For example, in the areas of predicting driver drowsiness, different computer vision based systems have been developed for monitoring driver vigilance by monitoring eye blink rate, head nodding etc [10]. Other research has shown the possibility to detect drunk driving patterns from smartphone sensors [4] and to monitor elders driving pattern through a sensor fusion of accelerometer, camera and GPS [1].

More directly relevant to our research is work that detects various driving events. Research has demonstrated the promise of classifying driving events (e.g. turn events, stop, start) through inertial sensors on a phone [5] [9] and the sensor data from a vechile [7] with different algorithms(e.g. DTW, HMM). Our work is similar in terms of sensing sources, but instead of identifying driving events, we distinguish between different potential drivers of the same car.

Finally, there is some work modeling driving behavior using sensed data but this makes different assumption about data availability. Some research built drivers' profile based on the additional sensors from the car, which is not widely available through OBD-II car ports [6] and others collected the sensors from smartphone to identify the driver vs. passengers [2]. The closest work is [8], which was able to classify different drivers, but it instrumented the brake and gas pedals with customized sensors sampling at high frequencies (1 kHz), which make it less practical to be applied in the real world.

Most of the research above focuses on driving event detection, using customized sensors or having access to proprietary sensors. We instead chose a more practical hardware setting in current real-world scenarios using the ODB interface and a smartphone. The OBD-II interface is available in every car produced after 1996 in the US and phones have been widely used as testbeds for sensor data collection [2] [4] [5] [6]. Thus the challenge we address is using the data from commodity hardware to build a driver classification model.

DATA COLLECTION

To build a driver classification model based on driving behavior, we need data from multiple drivers. We collected data in two different scenarios. In the first data set we collect data from 14 drivers who each drove 3 different cars on a predefined route. With this setup we wanted to understand the ability to classify a many drivers and their associated driving behaviors on a single car with a user-consistent driving task. The second data set was a more naturalistic driving setting where we recruited 3 couples who share a car and collected their everyday driving activities over a 4 day period (Friday to Monday). The data collected under this condition was less controlled, with ad-hoc routes, and hence would allow us to test the robustness of our model.

We equipped cars with an off-the-shelf Bluetooth OBD-II scanner that connects to a Samsung Galaxy S3 Android

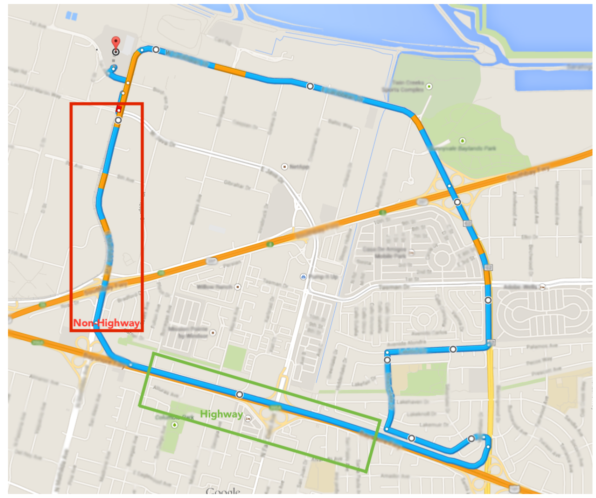

Figure 1: Predefined route followed by users during data collection (image blurred for anonymity purposes).

smartphone placed in the central console with a custom Android data logging application we wrote. In our Android logger application, we used the Torque API[2] which manages Bluetooth connectivity and the sensor data coming from the OBD-II scanner. We log all the sensor streams available through OBD-II scanner as well as the phone sensor data including accelerometer, gyroscope, magnetometer, GPS, temperature and the light sensor. We optimized the application to wake up only when the car is turned on and it would go to sleep after the car is turned off.

Procedure

For the first data set, we collected data from 14 (8 male and 6 females) participant drivers who were recruited using convenience sampling from our corporate research lab. All of the participants were asked to drive 3 different cars in a random sequence (2014 Honda Accord, 2013 Honda Accord, 2012 Scion Coup) on a pre-defined route as shown in Figure 1. The predefined car route was 7.2 miles (11.6 km) and included surface streets, freeways, ramps, traffic lights and

[2] https://play.google.com/store/apps/details?id=org.prowl.torque

Sensor & Sensor Location		Dims	Info. Gain
Gyroscope	phone	1	31.9
Torque	car	1	16.43
GPS	phone	1	9.14
Accelerometer	phone	1	8.89
Acc. pedal position D	car	1	0.62
Throttle position manifold	car	1	0.56
Absolute throttle position B	car	1	0.38
Relative throttle position	car	1	0.35
Acc. pedal position E	car	1	0.29
Engine RPM	car	1	0.09

Table 1: Sensors used to classify drivers, the dimensionality of the sensor output, and information gain from each sensor used for driver classification.

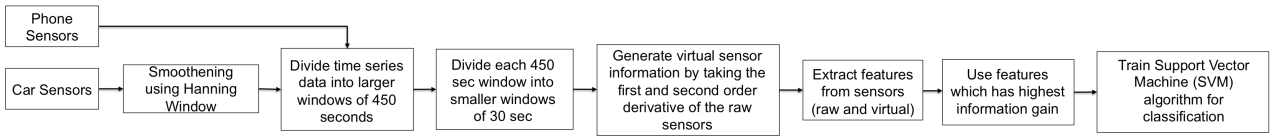

Figure 2: Block diagram used for classification modeling driver classification algorithm

stop signs. Each driver provided driving data for approximately 21.6 miles (34.8 km) in total (7.2 miles per route * 3 cars). The data collection was conducted over 3 days and between 10:30 AM to 4:30 PM. On average, participants took M=20.13 minutes (SD=3.90) to complete the course. Before driving, the participants were given instructions about the route and were asked to report back any deviations. Two participants redid the driving task as they had missed several key points on the route.

For the second dataset, we recruited 3 couples who share a car. The cars owned by the couples were of different makes (Volkswagon Jetta 2013, Honda Element 2004 and Audi A4 2012). As with first dataset, each car was equipped with the OBD-II dongle and a smartphone with our logging application. The smartphones were attached to an external battery so that the phone battery lasted 5-6 days. The couples were asked to keep a log sheet which contained the date, start time, end time and name of the driver so as to get the ground truth of who drove the car with approximate start and stop times so we could label the data collected. The 3 couples drove the car for totals of 203, 279 and 321 minutes with each driver driving the car for more than 1.25 hours.

DATA PROCESSING

Given our focus on driving behavior, we select a subset of sensors from Table 1. For example, the absolute throttle and accelerator provided information about how the gas pedal is pressed by the driver. From the smartphone, we log linear acceleration, gyroscope and acceleration information derived from GPS. The car sensor data from the OBD-II sensor is sampled at approximately 5 Hz whereas the gyroscope, linear acceleration sensor data and GPS data were sampled at 90 Hz, 90 Hz and 1 Hz respectively. We avoided using sensors and features that were related to road or traffic conditions (*e.g.,* instead of speed which is influenced by the traffic/road

Features	Dimensions
Histogram	10 (Car) 30 (Phone)
Entropy of Histogram	1
Power Spectrum	5 (Car) 26 (Phone)
Entropy of Power Spectrum	1
St. Dev. of Power Spectrum	1
Cepstrum	5 (Car) 26 (Phone)
Entropy of Cepstrum	1
St. Dev. of Cepstrum	1
Max, Min, Mean, Median, and Variance per car sensor	1

Table 2: Features extracted from all the sensors (virtual and real)

conditions, acceleration and jerk were used which are more closely related to driving behavior).

We used a Hanning window of size 11 to smooth the OBD-II sensor data. Further, we created virtual sensors by taking the first and the second order derivative of the physical sensor outputs. The time series data is divided into windows of 450s which are further divided into sub-windows of 30s each. The window sizes were optimized to accurately capture small driving changes without significantly compromising classification accuracy, using our first data set.

For each 30s window we extracted a total of 6400 statistical and frequency based features from the physical and the virtual sensors. We iteratively reduced the features based on the ranking of information gain and associated impact on accuracy. Afterwards, we verified the results of the feature pruning using 10-fold cross-validation. Using this process, we reduced the number of features to 1171. Table 2 provides details of the features and the dimensions related to each feature as provided to our model. A classifier was built to classify instances from each 30s sub-window; 15 instances (450s) were aggregated to vote for a driver. If there was a tie for different drivers based on votes/probabilities, we categorized it as a misclassified case. We used a support vector machine (SVM) model with a polynomial kernel with a cost function of one as our driver classification model.

EVALUATION

We performed an off-line evaluation of our approach with leave-one-out cross-validation. We performed three different experiments to test *a*) the model's performance on an artificially large number of drivers, *b*) the impact of driver count on model performance, and *c*) the robustness of the model in natural non-prescribed environments. The first dataset was used for Experiments *a* and *b*; the second naturalistic driving data was used for Experiment *c*.

In Experiment *a*, the SVM model classified the correct driver with an accuracy of 85.00%, 93.22% and 61.42% using only the phone sensors for car 1, 2 and 3 respectively (Table 3) which is far greater than a random chance classification (7.14%). We also tested the accuracy of the SVM model using only car sensor data and a combination of both car and phone sensors. The classification accuracies using only the car sensors were 37.14%, 25.36% and 28.57%, whereas the accuracies were 82.00%, 88.56% and 61.43% when classified using both the car and phone sensors.

In Experiment *b*, we examined the impact of the number of drivers on the classification performance. We used car 1 data from our first dataset and examine the impact on accuracy for between 2 and 14 drivers (Figure 3). The classification ac-

	Pre-defined route - Experiment a (Data set 1)				Naturalistic driving- Experiment c (Data set 2)			
	Car 1 Scion Coup (2012)	Car 2 Honda Accord (2014)	Car 3 Honda Accord (2013)	Mean Car 1,2,3	Car 4 Volkswagon Jetta (2013)	Car 5 Audi A4 (2012)	Car 6 Honda Element (2004)	Mean Car 4,5,6
Phone Sensor	85.00	93.22	61.42	79.88	67.50	75.00	97.50	80.00
Car Sensor	37.14	25.36	28.57	30.36	80.00	87.50	90.00	85.83
Car & Phone Sensor	82.00	88.56	61.43	77.33	62.50	82.50	97.50	80.83

Table 3: Accuracy attained using different datasets and using each sensor source independently and by combining them

curacy for the model using only the car sensor data shows a large drop once the number of drivers is greater than 3. However the classification accuracy was 100% for 2 drivers classification when tested with phone sensors, car sensors and a combination of both car and phone data (Figure 3).

As a final validation of our approach and to test the robustness of our approach in the real-world, we use our second naturalistic dataset, *which was not used in designing our approach*. We again tested the classification accuracy using only phone sensors, only car sensors and the combination of both car and phone sensors (Table 3). The model driver classification accuracy for each car-couple pair was 67.50%, 75% and 97.50% for cars 4, 5 and 6 respectively. Similarly, the classification accuracy using only the car sensors was 80%, 87.50% and 90% and using the combined car and phone sensors was 62.50%, 82.50% and 97.50% for car 4, 5 and 6 respectively.

DISCUSSION

Reflecting on these results, we see that only using the car sensors results in rather poor performance when there were more than 3 drivers. This result is evident in Experiment *a* (Table 3) for car 2 and 3 in particular as well as Experiment *b* (Figure 3). This could be caused by insufficient separation between drivers in the reduced feature space created by just the car sensor data.

The model can classify between a few drivers with higher accuracies using only phone sensors and a combination of car and phone sensors. The classifier gave the best accuracy when tested with fewer drivers. The classification accuracies were 100%, 100% and 100% (2 drivers, Figure 3) using the first dataset when tested with 2 drivers using only phone sensors, only car sensors and combined car and phone sensors.

The results of Experiment *c* shows our model performs well in real-world driving situations. We obtained accuracies of 75.83%, 85.83% and 86.67%. Together our experiments reinforce our model's capability to classify a small set of drivers with data collected in both controlled and naturalistic driving condition.

Lastly, analyzing sensors based on the maximum information gained for both car and phone sensors, we derived that features extracted from Torque sensor (16.43) of the car provided the most information whereas gyroscope sensor of the phone (31.9) provided the highest information (see Table 1). Other sensor information (GPS, accelerometer from phone; acceleration pedal position and throttle position from the car) also provides information necessary to model driving behavior. Reducing the number of features based on information gain not only reduced the overall training features by 82% but provided the same level, if not better, accuracy than using all the 6400 features.

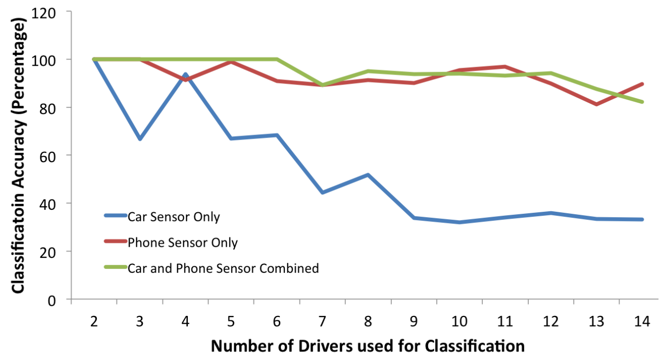

Figure 3: Classification accuracy by incrementally increasing the number of drivers in the model (Experiment b)

FUTURE WORK & CONCLUSIONS

Our aim was to develop and evaluate a driver classification model. We plan to extend this work in three directions. First, we plan to deploy the system to work online in real-time. Second, we plan to use our classification system to develop a variety of advanced mobile applications and/or services (*e.g.,* shift the driver's phone from touch interaction to hands-free mode) for more personalized in-car experiences. Third, we are looking at opportunities to improve our classification accuracies. For example, we are experimenting with a Bayesian network based learning approach (akin to [3]) which learns the importance and dependencies of each sensor.

In this paper, we presented a driver classification model inferred from driving behavior captured by low level sensors. The sensors consisted of car sensors available through the OBD-II scanner and smartphone sensors. The proposed model was tested with each set of sensors independently as well as by using the combined set. Different characteristics of the model such as ability to classify differing number of drivers under different driving conditions from prescribed to naturalistic driving were evaluated. While classifying a large number of drivers is challenging, we found the two driver case to perform well which opens to door for new applications.

ACKNOWLEDGMENTS

The authors would like to thank all the volunteers who took part in the data collection process and all the reviewers for their constructive feedback.

REFERENCES

1. Baldwin, K. C., Duncan, D. D., and West, S. K. The driver monitor system: A means of assessing driver performance. Tech. rep., Johns Hopkins APL Technical Digest, 2004.
2. Bo, C., Jian, X., Li, X.-Y., Mao, X., Wang, Y., and Li, F. You're driving and texting: Detecting drivers using personal smart phones by leveraging inertial sensors. In *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, ACM (2013), 199–202.
3. Chickering, D., Heckerman, D., and Meek, C. A bayesian approach to learning bayesian networks with local structure. *CoRR abs/1302.1528* (2013).
4. Dai, J., Teng, J., Bai, X., Shen, Z., and Xuan, D. Mobile phone based drunk driving detection. In *4th International Conference on Pervasive Computing Technologies for Healthcare* (March 2010), 1–8.
5. Johnson, D., and Trivedi, M. Driving style recognition using a smartphone as a sensor platform. In *14th International IEEE Conference on Intelligent Transportation Systems* (2011), 1609–1615.
6. Ly, M. V., Martin, S., and Trivedi, M. Driver classification and driving style recognition using inertial sensors. In *IEEE Intelligent Vehicles Symposium (IV)* (June 2013), 1040–1045.
7. Mitrovic, D. Reliable method for driving events recognition. *IEEE Transactions on Intelligent Transportation Systems 6*, 2 (June 2005), 198–205.
8. Miyajima, C., Nishiwaki, Y., Ozawa, K., Wakita, T., Itou, K., Takeda, K., and Itakura, F. Driver modeling based on driving behavior and its evaluation in driver identification. *Proceedings of the IEEE 95*, 2 (Feb 2007), 427–437.
9. Sathyanarayana, A., Sadjadi, S., and Hansen, J. Leveraging sensor information from portable devices towards automatic driving maneuver recognition. In *15th International IEEE Conference on Intelligent Transportation Systems* (2012), 660–665.
10. Zhang, W., Cheng, B., and Lin, Y. Driver drowsiness recognition based on computer vision technology. *Tsinghua Science and Technology 17*, 3 (June 2012), 354–362.