AUTHENTICATING DRIVERS BASED ON DRIVING BEHAVIOR

BY MAHRAD SALEMI

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Prof. Janne Lindqvist

and approved by

_____

_____

_____

_____

New Brunswick, New Jersey

May, 2015

ABSTRACT OF THE THESIS

Authenticating Drivers Based on Driving Behavior

by Mahrad Salemi

Thesis Director: Prof. Janne Lindqvist

Contemporary cars come by default equipped with several kinds of anti-theft systems. Despite this, the cars remain vulnerable to a simple physical attack: stealing the keys. Towards the end of improving anti-theft systems, we present the design, implementation, and evaluation of a driver behavior based authentication system. Our system uses various data available from the cars' computers as features in the authentication. For evaluating our approach, we collected driving data from 30 participants in a controlled experiment. Our approach shows promising results in distinguishing the authorized drivers from unauthorized drivers with a mean Equal Error Rate (EER) of 4.44%.

Acknowledgements

First and foremost, I would like to thank my adviser, Professor Janne Lindqvist, who provided crucial guidance throughout the course of this research. I also would like to thank all my colleagues at Rutgers ECE Department's Human-Computer Interaction group who helped me in data collection in the preliminary stage of the work.

Table of Contents

Abstract ii

Acknowledgements iii

List of Tables vii

List of Figures viii

1. Introduction 1

2. Related Work 4

2.1. Overview 4

2.2. Vulnerabilities of Car Systems 4

2.3. Automotive Anti-Theft Systems 4

2.4. Driving Style Recognition 5

2.5. Driver Authentication by Driving Behavior 6

2.6. Summary 8

3. Method 9

3.1. Overview 9

3.2. Participants 10

3.3. Apparatus and Materials 10

3.4. Procedure 11

3.5. Study Environment 12

3.6. Data Preprocessing 13

3.7. Summary 14

4. Preliminary Approaches 15
4.1. Overview 15
4.2. Wavelet Semblance Analysis 15
4.3. Spectral Analysis 17
4.4. Other Methods 19
4.5. Summary 20

5. System Design 21
5.1. Overview 21
5.2. Design Considerations 21
5.3. Design 23
5.3.1. Data Splitter 24
5.3.2. Feature Extraction 24
5.3.3. Feature Selection 28
5.3.4. Classifiers and Decision Threshold Selection 28
5.4. Summary 32

6. Results 33
6.1. Overview 33
6.2. Implementation run-time 33
6.3. Authentication Performance 34
6.4. Individual Analysis of Drivers 35
6.5. Analysis of Features Effect 36
6.6. Effect of Test Data on Authentication 37
6.7. Summary 38

7. Discussion 40
7.1. Overview 40
7.2. Factors Affecting Results 40
7.3. Failed Approaches 42

7.4. Summary 43

8. Conclusions 44

References 46

List of Tables

3.1. Table shows information derived from the car during data collection. . . 14

4.1. D stands for driver. For each driver, there are two driving traces: driving trace 1 and 2. The table matrix shows the common dominant frequencies for driving traces 1 and 2 for Engine RPM data. For example, in the first row, (D1,D1) have two common dominant frequencies which are greater than of those for (D1,D2), (D1,D3), or (D1,D4). Also, "-" means there are no common dominant frequencies. 19

5.1. Description of extracted features which are used for driver authentication. All features are extracted from Speed, Engine RPM, and Gyro data. 27

6.1. Average EER and STD values. Females have by about 1% less mean EER, however, the standard deviation in their group is higher. 34

6.2. EER values for 30 drivers. The EER for driver n means that all 30 drivers have claimed identity n and their driving data is compared against template driving profile n among which only one is authorized and others are unauthorized. 36

6.3. Table shows the combination of features which provides minimum EER for each n-F plot ($n = 1$ *to* 6) as appears in Figure 6.2. Features are shown with their corresponding number. 37

List of Figures

1.1. This thesis presents Driver Authentication System (DAS) which authenticates drivers based on their driving data. The figure shows driver authentication timeline after creating driving profile. After ignition, driver enters their ID, in other words, claims their identity. While driving, DAS collects driving data from ECU and processes and matches them against the claimed driving profile. If the driver is unauthorized, DAS instructs immobilizer to deactivate ECU, otherwise no action is required and ECU maintains its normal operation. 2

3.1. Preliminary Study. The figure shows the routes on which participants drove. This route was taken by participants twice round trip. The route characteristics include local areas with high traffic as well as highway with light or no traffic. It also constitutes several turning and stops events. 12

3.2. Formal Study. Route A shows the starting route from Busch to Cook campus which comprises mostly suburban areas with more traffic lights. Route B shows the return route from Cook campus to the starting point. This route includes mostly highways with light traffic conditions. 13

4.1. Figure shows Semblance for two data traces, in this case speed traces for two driving sessions by one driver for a specific route. Semblance compares two signals based on their phase angle as a function of frequency. Red shows positive correlation, green implies no correlation, and blue indicates anti-correlation. 17

4.2. Figure shows smoothed periodogram (spectral density) of RPM data for 4 different drivers. The black and red graphs belong to the same driver for 2 different driving sessions. It can be seen, dominant frequencies for both graphs in most of the cases match. This suggests the possibility of recognizing drivers by analyzing their driving data in frequency spectrum and finding their dominant frequencies. The dominant frequencies in driving data can be a distinguishing factor between drivers. 18

4.3. Figure shows two time series which have an overall similar shape but are not aligned in the time axis. The nonlinear dynamic time warped alignment allows a more intuitive distance measure to be calculated [1]. However, this method cannot be utilized for our purpose since speed traces differ in shape and pattern for different route characteristics. . . . 20

5.1. The Figure shows different components involved in driver authentication based on driving data. While driving, data flows from ECU to DAS Module where data processing and authentication is performed. If the driver is authorized, the immobilizer instructs the ECU to maintain its normal operation, otherwise it deactivates ECU. 22

5.2. Figure shows the correlation between the driving variables used for feature extraction. Pinkish color shows high positive correlation; Blueish color shows anti correlation and the light color shows weak correlation between the variables. The correlation coefficient between Engine RPM and Throttle Position is 0.69 and is the highest correlation coefficent between 2 variables among all. We chose to remove Throttle Position for further analysis since Engine RPM contains its information. 26

5.3. Figure shows sequence of all seven extracted features for three participants as a comparison example. The three sequences are more distinguishing in Maximum Speed and Average Calibrated Gyro while they are so similar and close in Average Breaking Speed. 29

5.4. The figure shows Greedy Forward algorithm for feature selection. Given a set of features, this algorithm finds the combination of features which gives the highest classification accuracy. In the figure, ΔS_{f_i} represents classification accuracy improvement and S_0 is the threshold to stop the algorithm if ΔS_{f_i} drops below that. Also, n_0 shows total number of features while k indicates number of selected features. 30

5.5. This Figure shows ROC curve for the fusion classifier. ROC is a measure of performance for binary classifiers. The more the curve is closer to the top-left corner of the box, the better is the performance. EER is found from ROC curve. 32

6.1. Figure shows histogram of EER values for 30 drivers. Each bar represents number of drivers who have the same EER range. 34

6.2. The figure shows EER values for different combination of features. (F stands for feature.) This means that each time different numbers of features are employed and authentication is performed and EER is calculated. For example, for 1F plot, all features individually are employed for authentication and EER is obtained, then all EER values are sorted in increasing order. In the horizontal axis, the Index shows the iteration for which EER is calculated. From the figure, as the number of employed features increases, the EER values decrease significantly from 1F to 3F plots but the improvement rate becomes slower afterwards. The lowest EER happens in 5F plot which means that employing 5 specific features results in lowest EER that is 4.44%. 36

6.3. The figure shows EER variation when input test data percentage to the system is varied from 50% to 100% of full test data (4.9 miles). EER improves significantly from 50% to 75%, then the improvement rate becomes slower until it stops at 95%. 38

Chapter 1

Introduction

Car theft remains a major problem around the world. Just in the United States, the Federal Bureau of Investigation estimates that final statistics for stolen vehicles for the year 2013 will be "just under 700,000 units. [2]" The estimated total losses are estimated to be over $4 billion a year [2]. A study conducted by Bureau of Justice Statistics [3] counted that only 42.6% of stolen vehicles were recovered in 2008 which cost a victim $7,821 on average in that year.

According to researchers [4], there are two types of temporary and permanent motor vehicle theft in the United States. The temporary thefts are typically committed by juvenile and young population for the purpose of joyriding or due to lack of access to the household vehicle. The permanent thefts are conducted by the more experienced and professional auto thieves who will make use of the stolen vehicle for variety of purposes including selling the spare parts, exporting vehicles across the borders or transportation. The stolen vehicles are not recovered in most of the reported cases which makes auto theft to have the lowest clearance rate among all federal offenses [5].

In this thesis, we present Driver Authentication System (DAS), which authenticates drivers using only their driving data. The system is robust and uses only driving speed and engine RPM from car's Electronic Control Unit (ECU) as well as gyroscope data available in contemporary cars [6]. In this work, we employed smartphone's inertial sensors to obtain gyroscope data. When a driver first time driving a car, they will create a new driving profile. During this first drive, DAS will learn the typical driving behavior of the driver. This profile is used for authenticating the driver during later drives. Figure 1.1 shows the process that the driver is authenticated assuming that they have a created a driving profile.

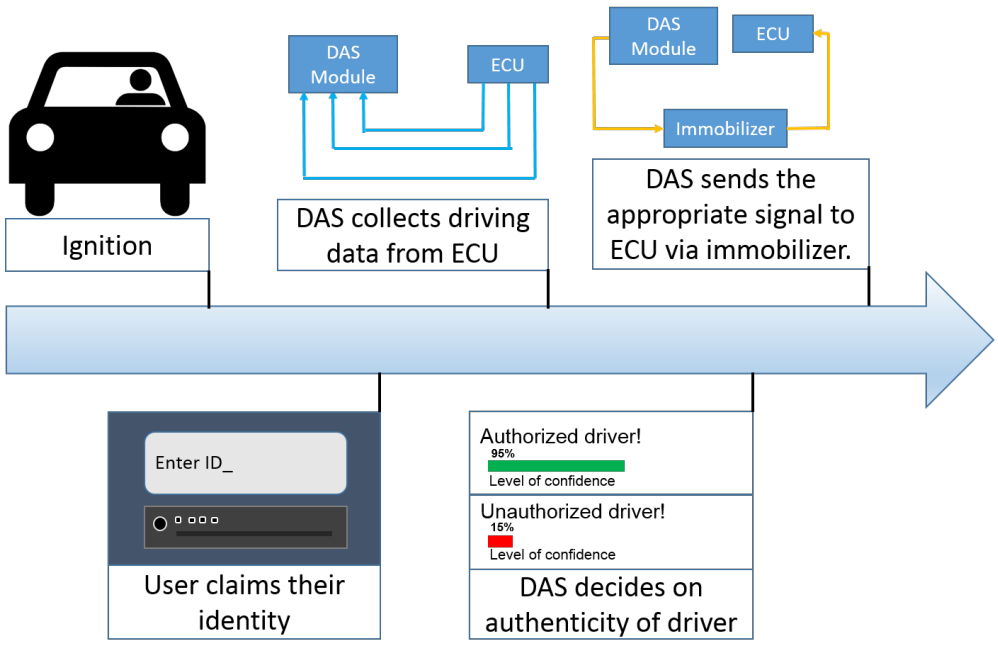


Figure 1.1: This thesis presents Driver Authentication System (DAS) which authenticates drivers based on their driving data. The figure shows driver authentication timeline after creating driving profile. After ignition, driver enters their ID, in other words, claims their identity. While driving, DAS collects driving data from ECU and processes and matches them against the claimed driving profile. If the driver is unauthorized, DAS instructs immobilizer to deactivate ECU, otherwise no action is required and ECU maintains its normal operation.

We conducted an experiment with 30 participants to test our system. We asked participants to drive a specific route twice using a test car. Many of them were not familiar with the route and were being given directions while driving. We used an Android mobile phone and a Bluetooth-enabled OBD-II device connected to the car to collect their driving data.

There has been some previous work on identifying drivers based on their driving behavior. One approach is to use inertial sensors of a smartphone to analyze acceleration, braking, and steering patterns [7, 8, 9, 10, 11]. We also leveraged inertial sensors to extract gyroscope data, however we mostly extracted our features from non-inertial sensors. Some studies have focused on features like distance from following car or foot gestures [12, 13] to distinguish between drivers. These information may also help for driver authentication but it is costly and not practical for widespread use.

Our goal is to introduce a system using basic available data from a car's computer

without a need for other external devices such as camera inside the car for extracting features from the driver's body gestures. This also ensures that any modern car equipped with OBD-II port could benefit from our system.

Our results show that the system performs well for typical driving conditions in an experimental setting.

Our contributions are as follows:

1. Report on driving data patterns collected from 30 participants;
2. Design of a driver authentication system based on basic driving data including speed, engine RPM, and gyroscope data as the primary inputs;
3. A preliminary study on driving behavior recognition using wavelet and spectral analysis and analyzing their weaknesses and strengths for our application.

Chapter 2

Related Work

2.1 Overview

In this chapter, we present the related work on vulnerabilities of car systems, automotive anti-theft systems, driving style recognition, and driver authentication by driving behavior.

2.2 Vulnerabilities of Car Systems

Checkoway et al. [14] studied the possibility of remote control attacks against cars. They demonstrate that a modern car's computer systems can be remotely compromised and fall into control of adversaries. Robinson-Mallet [15] gives a comparison of safety and security standards of ISO 26262 and ISO 15408 with respect to their application and possible misuse in automotive electronics. Rouf et al. [16] have studied the vulnerabilities of in-car wireless networks, in particular with Tire Pressure Monitoring Systems. Koscher et al. [17] showed the possibility of infiltrating into a car's ECU using standard OBD-II port and inserting a malicious software. The authors were able to completely control all functional units of a modern car through a wireless channel. Francillion et al. [18] studied relay attacks against passive key-less entry (PKES). Sagsetter et al. [19] have focused in particular vulnerabilities of electric cars.

2.3 Automotive Anti-Theft Systems

There are several approaches for trying to prevent and mitigate the theft of vehicles using software or hardware-based systems. Chen et al. [20] and Liu et al. [21] have proposed systems that can send the car's location to the owner with GSM. Busold et

al. [22] have proposed a car immobilizer based on NFC-enabled smartphones. They propose a security framework using a set of secure protocols implemented in NFC-enabled smartphones to overcome the shortcomings of conventional immobilizers. Guo et al. [23] proposed four software-based security layers built-in ECU to remotely gain access to the stolen vehicle. However, these systems can be vulnerable to various attacks. For example, Roel Verdult et al. have demonstrated the vulnerabilities in the design of Hitag2 [24] similar to the cloning of car's functional unit [25], a transponder tag used by automotive industry. They showed that the secret key can be recovered by their emulator in less than six minutes. Furthermore, these approaches are vulnerable to a really simple attack, stealing the car key. This can be done by, for example, parking valets.

2.4 Driving Style Recognition

Boonmee et al. [26] have used 2-axis accelerometers to detect bad driving behavior of buses. Similarly, Dai et al. [27] used a smartphone's built-in accelerometer to detect drunk driving events. Johnson et al. [28] and Eren et al. [29] have used inertial sensors to detect aggressive and non-aggressive driving behaviors. Paefgen et al. [30] have showed that smartphone measurements tend to overestimate anomalies in driving maneuvers. A Driving Habit Recognition Framework proposed by Dante Papada and Kathryn W. Jablokow [31] uses data from ECU to extract and cluster different driving patterns using a neural network algorithm. The system can successfully recognize the driving behaviors. Hsiao et al. [32] monitor driving behavior and capture dangerous maneuvers which are uploaded into cloud for tracking the drivers' behavior for traffic safety. Hong et al. [33] in a very recent work developed an Android application which provides feedback to the driver based on their driving data including those collected by the phone such as acceleration. Their model aims to identify aggressive driving and classify them in two groups of violation-class and questionnaire-class. In another work, Verwer et al. [34] used a restrictive type of Time Automaton (TA) known as Deterministic Real Time Automaton (DRTA) as a framework for modeling sequences of data captured from Engine RPM, speed, and fuel consumption sensors and then employed a

learning algorithm known as RTI+ developed by themselves for truck driver behavior classification. The purpose of their study was to detect bad driving behavior by truck drivers.

2.5 Driver Authentication by Driving Behavior

Van Ly et al. [7] have used inertial sensors built in smartphones to collect datasets for the purpose of driving events detection and classification and investigated a 2-class problem, in other words, differentiation between two drivers by performing Support Vector Machine (SVM) and k-mean clustering methods on the different vehicle maneuvers including acceleration, braking, and turning events. The scope of their work involves only driving comparison of two drivers.

In distinguishing drivers, driving states recognition is also investigated [35]. Wathanyoo Khaisongkram et al. first modeled and recognized driving states and then based on a probabilistic approach, they identified drivers based on the sequence of their driving states. Xiaoning Meng et al. [10] also leveraged driving data such as steering angle, acceleration, and braking to recognize and distinguish authorized drivers from unauthorized ones. They applied Fast Fourier Transform (FFT), Principal Component Analysis (PCA), and Independent Component Analysis (ICA) to the data and then employed SVM as the learning method and classifier. A different approach was taken in [11] to use driving data, for example, engine rpm, speed, acceleration, and braking signals to differentiate between drivers. They used different fixed and trainable methods for driver verification.

Meng et al. [36] have designed an experimental platform, a real time graphical simulator which provides normal driving controls including steering, brake pedal, and gas pedal. They used this simulator to collect data and capture the driving events of steering, braking, and acceleration and then employed Hidden Markov Model (HMM) as the learning algorithm on 7-class problem to dynamically model an individual's driving behavior and differentiate the drivers by similarity distance measurement. Similarly,

Oliver et al. [8] have designed a graphical user interface (GUI) in LabVIEW for calibrating the car signals, triggering the acquisition, and annotating the driving maneuvers as they take place. Using five sensor signals for capturing acceleration, braking, speed, steering angle, and gear position as driving events and HMMs and potential extensions (CHMMs) as training algorithms, they developed a system for driving behavior recognition and prediction for future automated cars. This is an interesting approach for modeling driving behavior, however, a driving simulator cannot fully imitate the events that may happen in real driving environment which may lead to biased results.

In a different work, C. Tran et al. [12] have leveraged foot gestures while driving as a physical characteristic to distinguish drivers using vision-based tools. They designed model using HMM algorithm as the recognizer which resulted in remarkable accuracy. Distance from the following car and gas and braking pedals operations are also analyzed in order to identify drivers in [13]. They focused on the analysis of those pedals' raw data collected from driving simulator as well as real car with and without including spectral analysis. Their result showed that spectral analysis of those data can improve the identification rate. In a similar work, parametric and non-parametric models were employed for the velocity, and other features mentioned earlier for both real and simulated driving environments [37]. Such approaches need external tools other than the car itself to model driving profiles which makes it less practical for widespread use.

Finally, in a most closely related research to our own work, A. Wahab et al. [9] have performed a thorough study for 30 drivers on driving behavior recognition using extracted features from accelerator and brake pedal pressure as the behavioral inputs and Gaussian Mixture Models (GMMs) for feature selection. Two fuzzy-neural-network-based systems called EFuNN and ANFIS were then used for driver authentication purpose. They have used mean of accelerator and brake pedal pressure signals and their derivatives only during stop and go regions. Their work seems to be a promising approach, however, there are some issues that we describe in following. The stop and go region according to their definition, is "the period when the vehicle moves off from a stationary position until the moment when the vehicle comes to a complete halt." This means that they have not considered the situations that the driver does not completely

come to stop. For example, in the street and highway traffic where the driver slows down by pushing braking pedal but does not completely halt. In addition, they provide no information about the route characteristics on which data are collected. Given these facts, their method fails in driver authentication if the thief drives in a path that does not include stop and go regions for a considerable amount of time unless the system finds they are unauthorized drivers in the initial acceleration from the first stationary state. Our system extracts the features based on the whole driving session except for idling times and from both inertial and non-inertial data to overcome this issue.

2.6 Summary

We provided literature review on vulnerabilities of modern anti-theft systems and described how they can be compromised by remote attacks and fallen into control of hackers. Also, we described related work on driving style recognition by driving data. These works showed how dangerous driving maneuvers or aggressive and normal driving habits can be recognized and subsequently feedbacks can be given to drivers. Finally, we talked about driver authentication by driving behavior. We mentioned several similar works who have employed driving data to authenticate drivers. However, some of them have leveraged some other features as well such as foot gestures or distance from the following car. We described differences of our work with previous research.

Chapter 3

Method

3.1 Overview

In this chapter, we explain our method of driving data collection. We describe a preliminary study and a formal study. Our goal is to extract driving characteristics from participants' driving data. We ask participants to drive a car as they do on a daily basis with their own car. They were not given any hints about purpose of the study in order not to affect their driving style. To understand if participants drive a route similarly in multiple rounds of driving, we ask participants to drive a specific given route twice without any considerable delay.

In our studies, we use a regular car which has OBD-II port without adding any new features or functionalities. OBD-II system is a capability available in cars in the United States since 1996 which provides access to diagnostic information and trouble codes as well as a list of vehicle parameters taken from ECU such as speed to monitor [38]. Also, the OBD-II port is located within three feet of the driver under steering wheel and does not require any tool to be revealed. The amount of information that this system can provide varies between old and new cars. For example, old cars which do not have Electronic Stability Control (ESC) unit, cannot provide acceleration and gyro data. ESC constitutes steering wheel angle, yaw rate, lateral acceleration, and wheel speed sensors which provide information about motion of the car [39]. In the United States, ESC has become mandatory for all cars manufactured since 2012 [6]. Therefore, implementation of our system does not require additional functionalities for any car manufactured since then.

The car that we use in our study does not have capability to provide motion information (acceleration and gyro.) Hence, we leverage accelerometer inside an smartphone

in our study to obtain necessary data related to motion of the car.

Next, we describe our participants, apparatus, procedures, study environment, and data preprocessing for both studies.

3.2 Participants

Preliminary Study. We made an announcement via email list for our study. Four students from Rutgers University (two women and two men) volunteered to participate in the experiment. Three were graduate students while one was undergraduate. We only required the participants to have driver license.

Formal Study. We recruited 30 participants (18 male, 12 female), all affiliated with Rutgers University, by using online social media. We required the participants to be 18 years old or over and have a valid driver license and liability coverage insurance. Also, participants were required to be an active driver during the past three months when they participated in the study. Their educational background varies, however, most of them were undergraduate students while six had or were pursuing graduate degree. In this study, informed consent were obtained from all participants.

All participants completed the two driving sessions in our one session study which took about 1.5 hours for each participant. As compensation, they were given $50 gift card for their participation in the entire study.

We recruited participants in two batches: first in December 2014 (20) and second in January and February 2015 (10).

3.3 Apparatus and Materials

Common Apparatus in both studies. The driving data were recorded using a 2002 Hyundai Accent car, a Bluetooth-enabled OBDII adapter, a Samsung Galaxy SII smartphone, and an Android application called Torque. It is worth mentioning that OBD-II adapter receives its power supply from the car's battery.

Formal Study Materials. We constructed a short demographic survey that contained these questions: number of accidents during past year, gender, ethnicity, age

range, number of vehicles in the household, number of driving sessions on average per day, driving mileage on average per day, household income, and occupation. Research participants were presented the survey online by Google form.

3.4 Procedure

Formal Study. We asked participants to complete a short demographic survey. Next, we verified their driver license and liability coverage insurance and asked them to read the consent form. We also explained the driving risked involved to participants verbally as appeared in the consent form. Then, we described the whole process and showed them the driving routes. However, they were told that they will be given direction during driving if needed.

Common Procedure in both studies. We told participants that the study is being conducted to understand driving. We informed the participants that they should drive the car twice for a given route. They were shown the route prior to start of the driving. We connected OBD-II adapter to the OBD-II port inside the car under steering wheel. Also, we attached the smartphone using a holder on the windshield, long front side facing toward the seat. After ignition, we enabled the smartphone's Bluetooth connection and then by going to Torque application, the phone auto connected to ECU via OBD-II adapter.

For driving task, at first, participants were given time to drive with the test car around Busch Campus in order to get acclimated with the vehicle. Before starting the data collection, we calibrated the accelerometer of the smartphone while on the holder in the Torque application. Then, when they were ready, we started data collection and asked each participant to drive the given route.

Preliminary Study. We told participants to drive a given route and return to origin via the same route totally driving 3 miles in one driving session. Then, we asked them to repeat the driving task for the second time. Each participant drove 6 miles.

Formal Study. We instructed participants to drive from origin to destination via route A and return to origin via route B totally driving 9.8 miles in one driving session.

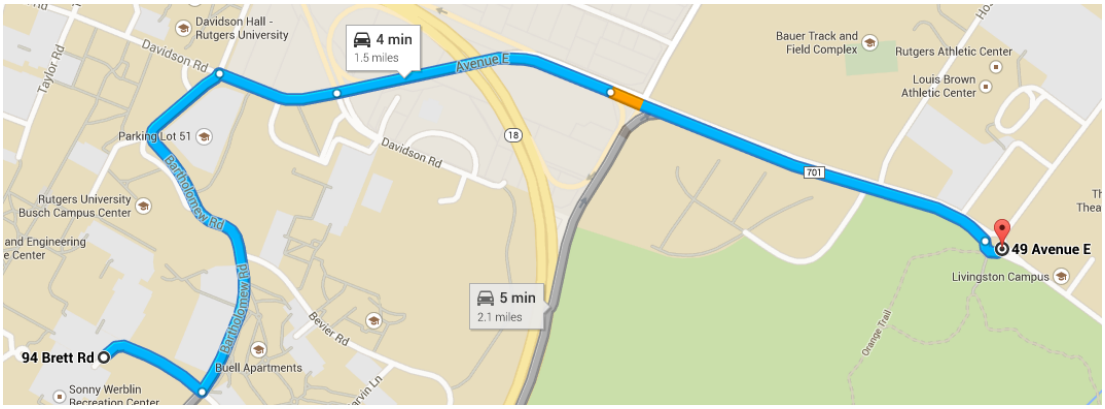


Figure 3.1: Preliminary Study. The figure shows the routes on which participants drove. This route was taken by participants twice round trip. The route characteristics include local areas with high traffic as well as highway with light or no traffic. It also constitutes several turning and stops events.

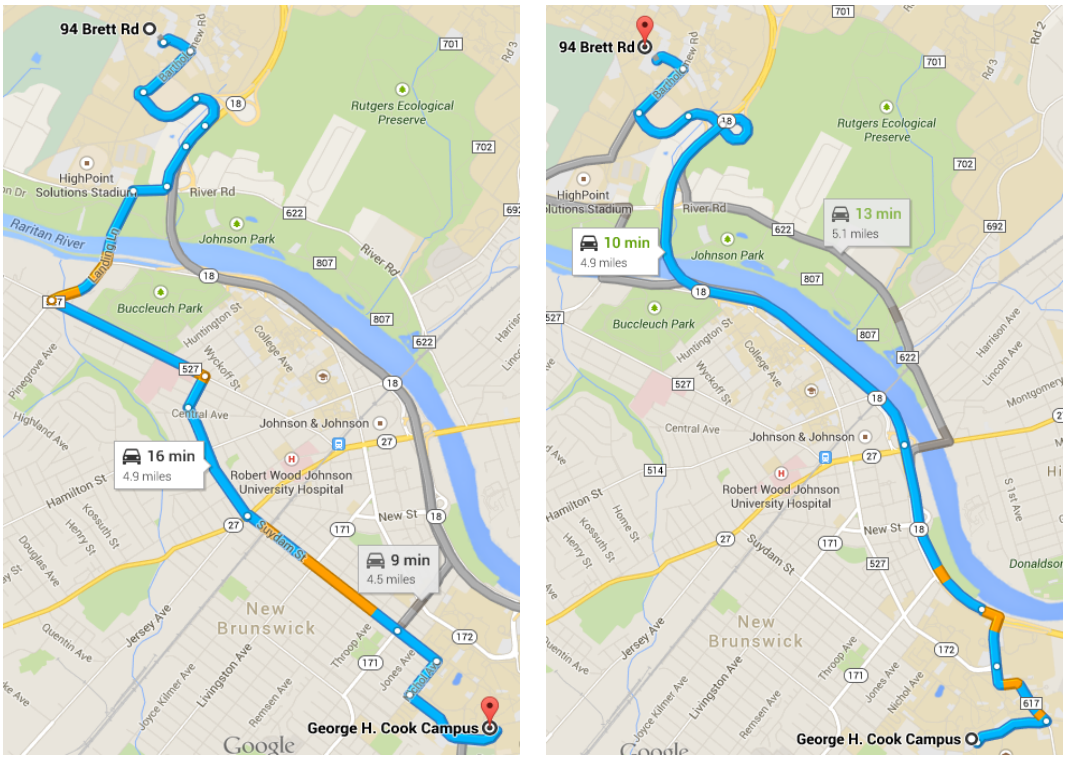
Then, we asked them to repeat the process. Each participant drove 19.6 miles.

The route characteristics for both studies are described in the next section.

3.5 Study Environment

Preliminary Study. We conducted the study in a summer day in June 2014 from 12pm to 3pm. We chose a 1.5 miles route between Busch and Livingston campuses of Rutgers University to conduct the experiment. The route characteristics include both streets with traffic and highway with less or no traffic. Also, it includes several turning events. Figure 3.1 shows the route on which the experiment was conducted.

Formal Study. We chose late morning and early afternoon to conduct the study to avoid rush hours as well as having the same traffic conditions. It is worth to note that weather condition was also similar for all participants. The study took place in an environment consisting of routes A and B. Route A included only local streets with more traffic whereas route B mostly included highways with light or no traffic. By choosing two completely different routes, we ensured that most driving characteristics are examined during the experiment. These characteristics include but are not limited to constant braking and acceleration in congested traffic condition, driving with high speed while changing lane, and turning maneuvers. Each route is about 4.9 miles which in total is 9.8 miles round trip. Figure 3.2 shows routes A and B.



(a) Route A (b) Route B

Figure 3.2: Formal Study. Route A shows the starting route from Busch to Cook campus which comprises mostly suburban areas with more traffic lights. Route B shows the return route from Cook campus to the starting point. This route includes mostly highways with light traffic conditions.

3.6 Data Preprocessing

The logged driving traces were then retrieved after the study was over and named by a number representing ID of the corresponding participant and stored for later analysis. Also, all available information were obtained from both car and smartphone. Table 3.1 shows all these information.

Preliminary Study. Totally, eight driving traces were acquired for four participants. The sampling rate was set to 1 Hz in this study.

Formal Study. In all, 61 driving traces were generated for 30 participants. An extra driving trace is due to an error in choosing the route by one of the participants which required us to repeat the experiment. The sampling rate was set to 1.43 Hz (higher than that of preliminary study) to capture more changes in driving data.

Parameter ID	Parameter ID	Parameter ID
Longitude	Latitude	Altitude
GPS Speed (ms)	Speed (OBD)(mph)	Bearing
G(x)	G(y)	G(z)
G(calibrated)	Engine RPM (rpm)	Acceleration Sensor (Total)(g)
Acceleration Sensor (X axis)(g)	Acceleration Sensor (Y axis)(g)	Acceleration Sensor (Z axis)(g)
Average trip speed (whilst stopped or moving)(mph)	Engine Coolant Temperature(Â°C)	Engine Load(%)
Horizontal Dilution of Precision	Fuel Trim Bank 1 sensor 1 (%)	Fuel Trim Bank 1 Short Term(%)
Fuel Trim Bank 1 Long Term (%)	Intake Manifold Pressure (psi)	Miles Per Gallon (Long Term Average)(mpg)
Miles Per Gallon (Instant)(mpg)	Throttle Position (Manifold)(%)	Trip Distance (miles)
Turbo Boost & Vacuum Gauge (psi)	Trip Time (Since journey start)(s)	Trip time (whilst moving)(s)
Trip time (whilst stationary)(s)	Timing Advance (Â°C)	Intake Air Temperature (Â°C)

Table 3.1: Table shows information derived from the car during data collection.

3.7 Summary

We explained our method for driving data collection. First, we conducted a preliminary study for four drivers and a short route. We employed the collected data for preliminary approach and experimentations. Also, the preliminary study provided the necessary data for system design which is described later in Chapter 5. Moreover, we conducted a formal study with 30 participants and asked them to drive a given route twice. The timing and weather condition of the experiment was almost the same for all participants.

Chapter 4

Preliminary Approaches

4.1 Overview

In this chapter, we talk about our preliminary approaches that we have tried to model driving profiles. We employ similarity measures between driving traces particularly speed and engine RPM. We believe these data should have similar patterns for multiple driving sessions of one driver but different patterns when comparing two distinct drivers. To this end, we employ similarity measurement techniques such as DTW to compare two time sequences or other methods performing time-frequency measurements such as Wavelet Analysis. In addition, we transform time domain data into frequency domain and measure similarities in frequency spectrum since we expect frequency domain may provide useful information about unique driving characteristics not present in time domain.

In following, we describe Wavelet Semblance Analysis as a potential method to compare driving data, particularly speed data. Next, we apply Spectral Analysis to engine RPM data to show whether it is possible to find driving characteristics in frequency domain. We also try Dynamic Time Warping (DTW) to measure similarities between two speed traces.

4.2 Wavelet Semblance Analysis

Unlike Fourier Transform which only explains frequency content of a signal and provides no information about its temporal resolution, Wavelet Transform takes the advantage of both temporal (spatial) and frequency domains and thus analyzes the signal with different frequency and time resolution for any given scale. Therefore, wavelet analysis

is a time-frequency transformation which enables us to not only find the frequencies of interest but also more or less spot the time intervals in which they occur. In this section, a wavelet-based semblance analysis is introduced. Semblance filtering compares two signals based on their phase angle as a function of frequency [40]. The Fourier Transform of $H(f)$ of a signal $h(t)$, is given by:

$$H(f) = \int_{-\infty}^{-\infty} h(t) e^{-2\pi j f t} dt \tag{4.1}$$

Where f is the frequency and t is the time. Usually the function $H(f)$ consists of real and imaginary parts or amplitude and phase. The latter is more applicable in this case as semblance makes comparison of phase angles at a given frequency. Hence, the difference between the phase angles of two data sets can be computed as given in equation 4.2 based on their calculated Fourier transforms.

$$S = cos\theta(f) = \frac{R_1(f)R_2(f) + I_1(f)I_2(f)}{\sqrt{R_1^2(f) + I_1^2(f)\sqrt{R_2^2(f) + I_2^2(f)}}} \tag{4.2}$$

Where $R(f)$ and $I(f)$ are the real and imaginary components of Fourier transform for both data sets as function of frequency respectively. The semblance S can take on values from -1 to +1. A value of +1 indicates perfect phase correlation between the two data sets. Also, the value of 0 implies no correlation and likewise -1 implies perfect anti-correlation between phases. Figure 4.1 illustrates comparison of semblance for two speed traces belonged to one subject for one route.

Figure 4.1 shows that there are perfect correlations at the beginning and ending parts of two speed traces while in the middle parts they are mostly anti-correlated. Also, somewhere between wavelengths 50 and 100 corresponds to most positive correlation between the two traces. This method suggests a viable similarity measurement technique when two signals are similar in pattern but vary by wavelength. So, when the same driver takes different routes with different speed patterns, this method is impractical.

Another shortcoming of this method could be that the sample size must be equal for the two traces being compared. This makes the Wavelet-based Semblance analysis

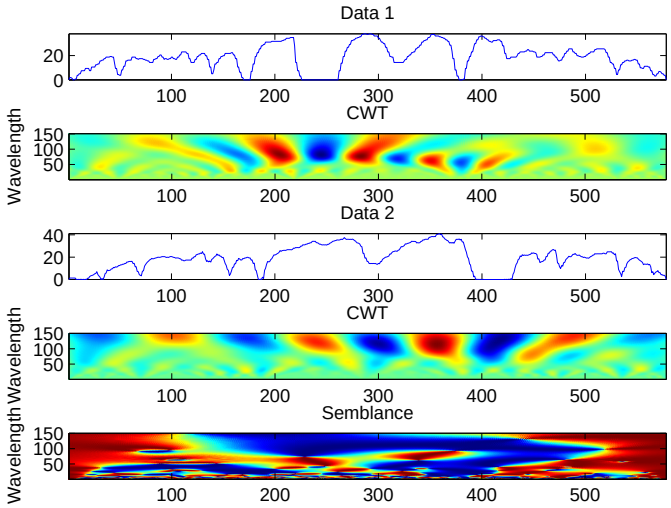


Figure 4.1: Figure shows Semblance for two data traces, in this case speed traces for two driving sessions by one driver for a specific route. Semblance compares two signals based on their phase angle as a function of frequency. Red shows positive correlation, green implies no correlation, and blue indicates anti-correlation.

less applicable for our purpose. However, it provides insights on how driving patterns may change when a driver drives the same route several times.

4.3 Spectral Analysis

In this approach, we only analyzed driving data in frequency spectrum. Driving data are time series which can be expressed as a combination of cosine (or sine) waves with different time periods and amplitudes, in other words, different harmonics. This fact can be utilized to examine the periodic (cyclical) behavior of time series. A periodogram is used to identify the dominant periods (frequencies) of time series. Periodogram can be considered an estimate of spectral density. Mathematically, the spectral density is defined for both negative and positive frequencies. However, due to symmetry of the function and its repeating pattern for frequencies outside the range -1/2 to +1/2, we only need to be concerned with frequencies between 0 and +1/2. We found spectral density of driving data, particularly for RPM variable to see if we could find useful information in frequency domain. We analyzed the spectral density for two driving sessions for 4 drivers.

Figure 4.2 shows the Engine RPM smoothed periodograms. From the figure, we could identify the dominant frequencies for each driver. The spectrums are shown for two RPM data traces together in order to see the similarities in dominant frequencies.

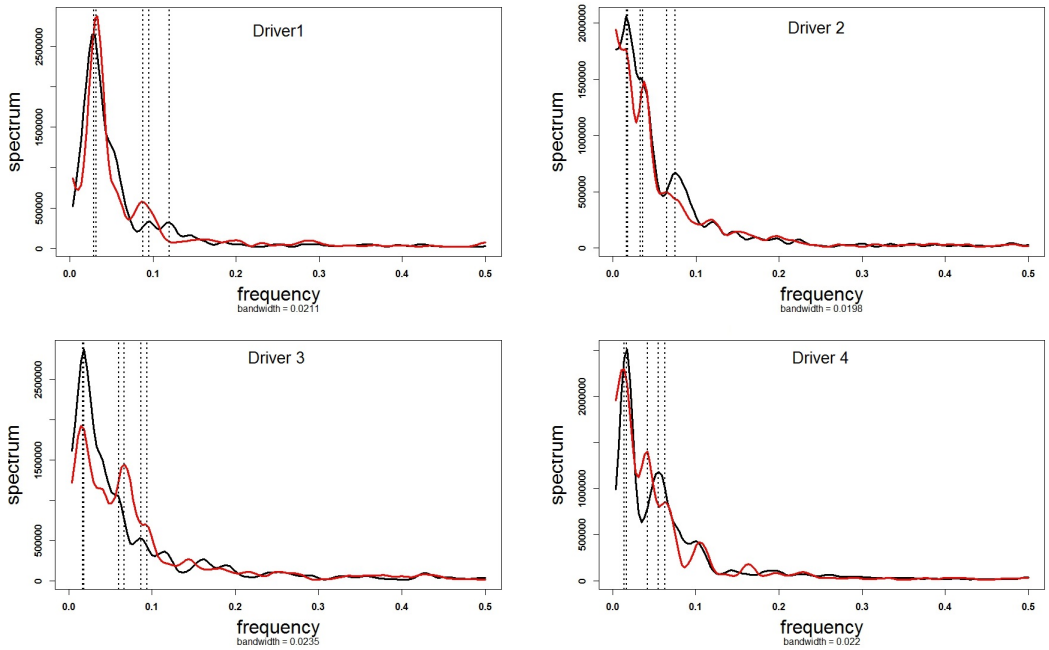


Figure 4.2: Figure shows smoothed periodogram (spectral density) of RPM data for 4 different drivers. The black and red graphs belong to the same driver for 2 different driving sessions. It can be seen, dominant frequencies for both graphs in most of the cases match. This suggests the possibility of recognizing drivers by analyzing their driving data in frequency spectrum and finding their dominant frequencies. The dominant frequencies in driving data can be a distinguishing factor between drivers.

Table 4.1 gives values of common dominant frequencies (CDFs) for two RPM data traces. We define a CDF between two traces as (f_{d_1},f_{d_2}) such that $|f_{d_1} - f_{d_2}| \leq 0.01$ which corresponds to $\frac{0.01}{0.5} \times 100 = 2\%$ tolerance since frequency axis expands from 0 to 0.5. We only compare RPM data in driving session one with RPM data in driving session two. RPM data within the same task are not compared to each other. We are interested in number of CDFs when comparing two RPM data. According to the table, number of CDFs are greater in (D_i,D_i) ($i = 1$ *to* 4) cases compared to (D_i,D_j) ($i = 1$ *to* 4, $j = 1$ *to* 4) ones. Having more CDFs shows the two traces should belong to one driver which in this analysis they do. This shows that a driver can be recognized by analyzing the picks in their RPM data's spectral analysis.

	Driving Trace 1			
Driving Trace 2	D1	D2	D3	D4
D1	(0.032,0.029) (0.088,0.095)	(0.032,0.033)	(0.088,0.087)	-
D2	(0.036,0.029)	(0.016,0.018) (0.036,0.033) (0.065,0.075)	(0.016,0.018) (0.065,0.060)	(0.016,0.017) (0.065,0.055)
D3	(0.094,0.095)	(0.017,0.018) (0.067,0.075)	(0.017,0.018) (0.067,0.060) (0.094,0.087)	(0.017,0.017)
D4	-	(0.014,0.018) (0.042,0.033)	(0.014,0.018) (0.063,0.060)	(0.014,0.017) (0.042,0.055) (0.063,0.055)

Table 4.1: D stands for driver. For each driver, there are two driving traces: driving trace 1 and 2. The table matrix shows the common dominant frequencies for driving traces 1 and 2 for Engine RPM data. For example, in the first row, (D1,D1) have two common dominant frequencies which are greater than of those for (D1,D2), (D1,D3), or (D1,D4). Also, "-" means there are no common dominant frequencies.

From this analysis, we would like to know whether drivers have certain driving characteristics in frequency domain. We tried to show this by finding dominant frequencies in their RPM data. However, having variable dominant frequencies for different drivers makes this approach difficult to implement.

4.4 Other Methods

We also employed Dynamic Time Warping (DTW) in order to match driving data. In DTW, a time series signal is warped in time to measure its similarities with a reference signal. For example, we measured the similarities between two speed traces. However, this method requires the two temporal sequences to have the same pattern or shape, in other words, they need to belong to the same driving route. Figure 4.3 illustrates this point. But, we need a system which can recognize and authenticate drivers regardless of their taken route which makes DTW not suitable for this application.

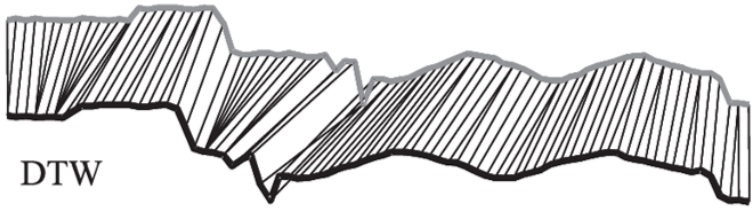


Figure 4.3: Figure shows two time series which have an overall similar shape but are not aligned in the time axis. The nonlinear dynamic time warped alignment allows a more intuitive distance measure to be calculated [1]. However, this method cannot be utilized for our purpose since speed traces differ in shape and pattern for different route characteristics.

4.5 Summary

We discussed our preliminary approaches to model driving profiles. We showed that Wavelet Semblance Analysis can only measure similarities of two time domain sequences which have similar patterns but are out of phase similar to DTW. For example, we analyzed two speed data traces for one route belong to one driver. Obviously, the speed pattern of one driver for several driving sessions for one specific route are similar but they may be out of phase. However, our purpose is to recognize a driver regardless of their taken route. We showed this method is not applicable to our goal.

We also tried Spectral Analysis, analyzing driving data in frequency domain. In spectral analysis of RPM data, we realized different drivers provide pick values at different frequencies. Also, the number of these pick values differ from one driver to another. This approach shows an interesting point in classifying different drivers according to frequency spectrum of their driving data and also number of pick values and dominant frequencies, in other words, where those picks occur. However, the complications involved makes this approach subtle to implement.

Chapter 5

System Design

5.1 Overview

In this chapter, we present a fusion method combing the results of a strong supervised learner and an ensemble learner. We describe our assumptions when designing the system. Modeling driving profile and driver authentication based on that model is inherently a complex problem since it involves not only personal behavioral traits but also many external factors. We present some design considerations and simplify the problem by neglecting some factors.

Figure 5.1 shows different components of system architecture. After ignition, data from ECU flows to our DAS Module where data processing and recognition is performed. Once DAS collects enough driving data while the driver is driving, it processes and matches those data with the claimed driving template. Then, based on that, it produces an authentication key and sends it to immobilizer in the car. The immobilizer then sends an authorization key to ECU depending on the DAS's outcome. If DAS sends a false signal, immobilizer deactivates ECU, otherwise no action is required and immobilizer instructs the ECU to maintain its normal status.

5.2 Design Considerations

We have considered five scenarios in real world that needs to be addressed when implementing the driver authentication system in the car's computer system. These situations are summarized below.

1. **Number of people:** A car can have one, two, or more drivers which is common in some families. In this situation, there needs to be the possibility to create

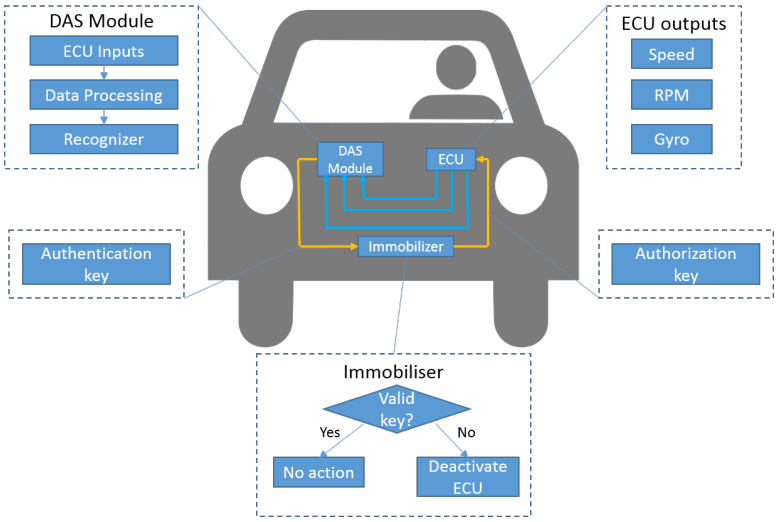


Figure 5.1: The Figure shows different components involved in driver authentication based on driving data. While driving, data flows from ECU to DAS Module where data processing and authentication is performed. If the driver is authorized, the immobilizer instructs the ECU to maintain its normal operation, otherwise it deactivates ECU.

driving profile for all authorized drivers.

2. **Lending car:** The owner lends their car to someone else, a friend for example or a customer who wants to rent a car. In this situation, the other person who doesn't own the car would be considered unauthorized if they drive the car without prior knowledge of the system that this person is authorized to in fact drive the car. Therefore, one possible solution especially in the case of rental cars is to define a mode to create a temporary authorized profile for the designated person(s). The temporary authorization will allow the customer to drive for a while to create the profile and then automatically switches to the authentication mode and also it could have a timing which could notify the owner should they forget or refuse to return the car on time.

3. **Variable driving pattern:** In some occasions, an authorized driver drives the car in an unusual deviation from the normal driving behavior. For example, the person is in hurry or drunk when driving the car.

4. **Environmental effect:** Weather and road condition affects the driving behavior. For example, driving in a clear weather is undoubtedly easier than driving in a

foggy weather.

5. **Regional effect:** The driver drives in a region which significantly makes the driving behavior norm different than their hometown. For example, the driver has usually driven in New Jersey but decides to travel to New York City and thus the city traffic condition and possibly unfamiliarity with the area might result in a deviation from the driving behavior norm.

Besides real world considerations, there are also a technical concern that needs to be taken into account. When the authorized drivers of a particular car decide to use the DAS for the first time, they need to create profile and drive for a while so that the system can collect their driving data and recognize them in subsequent driving sessions. However, the profile creation step itself must be password protected and secure so that some third party other than authorized drivers could not access to it to create driving profile. Conventional methods like text based password may become compromised and may put under question the whole effort for building such authentication system. On the other hand, accessing to create profile option should not be a hassle. Therefore, a secure mechanism is required to protect the system and at the same time it should not make user experience hard. But, this challenge is not focus of this research.

Also, we have designed our system without considering *variable driving pattern*, *environmental effect*, and *regional effect*. In other words, we assume the drivers always drive normally and in a normal weather and road conditions. In following, we describe all design steps.

5.3 Design

The preliminary collected data did not provide enough samples. Therefore, we used Multiple Imputation method to create simulated data and increase our sample size. Multiple Imputation is an algorithm which is primarily used for dealing with missing data [41]. However, this method can also be effective in creating more data similar to our existing data. We devised a method inspired by the work of Rubin [42] which views all data from nonsampled units to be filled by multiple imputation. We randomly and

intentionally removed a percentage of whole existing dataset to create a random pattern of missing data and then we applied multiple imputation to the dataset with missing values in five iterations. Then, we aggregated all imputed values together to produce a new final value for each missing value based on distribution of current data. Therefore, a new dataset can be created with the same sample size. Then, we concatenated the original and simulated data and used it for analysis. We only used this data extension method in order to test our system initially. In below, we describe different stages of our system.

5.3.1 Data Splitter

Each driving trace belonging to a driver has different sample size. Data Splitter algorithm divides the original driving trace to several traces so that each of which has enough sample size in order to be used for feature extraction. For instance, when a person is driving for 10 minutes (for example 600 data points for a sample rate of 1 Hz), some driving behavior such as speeding or braking patterns may repeat every 100 seconds which means that the whole driving trace can be divided to 6 driving traces. We use each of these driving traces for feature extraction. Furthermore, each of these driving traces corresponds to one sample size in the new reduced dataset.

We input a driving dataset and the expected length of a subset to the algorithm. It then finds the number of equal-sized subsets which gives more or less the same sample size given as the input for each subset. The pseudocode for this algorithm is given in Algorithm 1.

5.3.2 Feature Extraction

This section describes the feature extraction process from the three primary inputs: Speed, Engine RPM, and Gyroscope data. Figure 5.2 shows an example of the correlation between these variables for one of participants. The correlations just show similarities between driving variables and are similar for all participants. We could observe that there is a fairy high correlation between Speed and Engine RPM and also

Algorithm 1 Pseudocode for data splitter algorithm. Given length of driving subsets, the algorithm divides the main driving trace to equal-sized subsets.

Input: A driving dataset, Set and expected length of each trace, TL
Output: A list of subsets, Subsets and their lengths, L

```
1: procedure MyProcedure
2:     Size ← sample size of Set
3:     m ← round(Size/TL)
4:     r ← remainder(Size/TL)
5:     q ← (Size − r)/TL
6:     if m >= q then
7:         n ← q
8:     else
9:         n ← m
10:    L ← array(0)
11:    s ← 0
12:    for i = 1 to n do
13:        L[i] ← round((Size − 1)/(n + 1 − i))
14:    k ← length of L
15:    Subsets ← ∅
16:    for i = 1 to k do
17:        Subsets[i] ← Set[(s + 1) : (s + L[i])]
18:        s ← s + L
       return (Subsets, L)
```

between Engine RPM and Throttle Position. The correlation coefficient between Engine RPM and Throttle Position is the highest ($r = 0.69$). Throttle Position is a sparse vector which only contains non-zero values when the driver pushes the acceleration pedal. However, Engine RPM also includes those information and we chose to remove Throttle Position for further analysis.

Success of a classification algorithm heavily depends on the selected features in a data set. Hence, selecting good features for discriminatory purposes among different driving behaviors is crucial. Several application-dependent features are manually defined to extract relevant information pertaining to driving behavior recognition. These features include average speed, maximum speed, average braking speed, standard deviation of engine rpm, standard deviation of speed, average calibrated gyro, and another defined feature called rpm coefficient. Table 5.1 gives the description of extracted features.

RPM Coefficient Engine Revolutions per Minute (RPM), otherwise known as

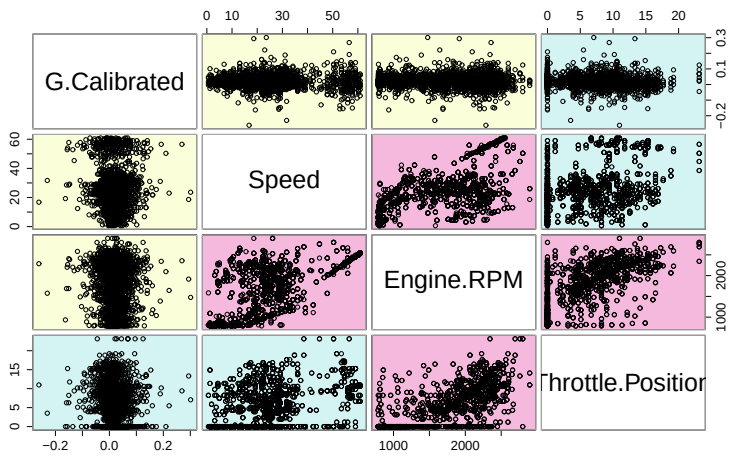


Figure 5.2: Figure shows the correlation between the driving variables used for feature extraction. Pinkish color shows high positive correlation; Blueish color shows anti correlation and the light color shows weak correlation between the variables. The correlation coefficient between Engine RPM and Throttle Position is 0.69 and is the highest correlation coefficent between 2 variables among all. We chose to remove Throttle Position for further analysis since Engine RPM contains its information.

engine speed, is the number of rotations completed in one minute around a fixed axis. Engine RPM and gear position have a direct relationship and change proportionally during a drive. When driving in a particular gear position, as the speed goes up, so does the RPM and when it reaches to a maximum value, either the gear levels up to a higher position in automatic transmission or the driver must shift the gear in manual one and the RPM drops afterwards.

One of the interesting factors in driving behavior recognition is monitoring changes in Engine RPM while driving. The goal is to determine how these changes in RPM would reflect human driving behavior as one factor among others. Therefore, we introduce a factor called RPM Coefficient for a particular driver and vehicle which could help in distinguishing different drivers driving a specific vehicle. This criterion states that how one is driving consistently in terms of pushing acceleration pedal in various situations. The formal definition is as follows:

$$RPMCoeff = \frac{STDEV(rpm)}{RANGE(rpm)} \tag{5.1}$$

The definition in 5.1 states that RPM Factor is the standard deviation of Engine

Feature	Description
RPM_COEFF	Standard deviation of engine rpm divided by the range of engine rpm.
AVG_BRK_SPD	Average of speed data points in monotonically decreasing periods.
AVG_SPD	Average of all speed data points.
RPM_STD	Standard deviation of engine rpm.
SPD_STD	Standard deviation of speed.
MAX_SPD	Maximum speed.
AVG_CALIB_G	Average of calibrated gyro data acquired from smartphone.

Table 5.1: Description of extracted features which are used for driver authentication. All features are extracted from Speed, Engine RPM, and Gyro data.

RPM divided by the range[1] of RPM values. As the standard deviation and range are both expressed in the same unit as data set measurements, the RPM Coefficient is unit-less. Also, this index always gives a number between 0 and 1.

Average Braking Speed: This variable takes the average of all speed data points in monotonically decreasing periods. For instance, these events may include the braking events in which the driver needs to stop for the red light before reaching to the intersection or other events that driver stops for road bumps or people crossing the street. This feature is chosen since it could be potentially useful in driving behavior recognition by examining alertness of the driver with respect to objects on the road or red lights.

Average Speed: This feature simply takes the average of all speed data points. The average speed, however trivial, is also a good feature representing that on average how fast the driver usually drives the car given the fact that people usually commute to limited places in a typical week. In other words, when going from residential places to work or to grocery stores, shopping malls etc, they take similar routes based on habit with more or less the same speed patterns.

RPM Standard Deviation: Standard deviation of engine rpm is similar to RPM_COEFF in some ways, however, it is not scaled by the range of rpm. Also, this feature indicates the pattern and intensity of acceleration pedal by the driver.

[1]Range= Maximum-Minimum

Speed Standard Deviation: This feature also shows the pattern of speeding and braking and how the driver is controlling the car's speed especially in uphill and downhill roads where the data from engine rpm does not disclose the true pattern because of force by gravity component.

Maximum Speed: Maximum speed is also a simple but important factor especially in regular commuting routes that the driver takes, since it can indicate approximately how fast the drive usually drives the car.

Average Calibrated Gyro: Gyro data is the only data which is acquired from inertial sensors. This data indicates the yaw rate and can help as a distinguishing factor for different drivers. Rate of change of motion of the car in turns, ups and downs, and bumps are recorded by Gyro sensor.

Figure 5.3 shows an example of the variation of the seven input features into the system for first three participants. As we can see, the values for "Maximum Speed" and "Average Calibrated Gyro" features are the most distinguishing factors in the case of these three drivers compared to "Average Braking Speed" feature, in which the values for the 3 people are so close.

5.3.3 Feature Selection

A subset of features are selected out of all the extracted features in this step. Feature selection follows a greedy forward algorithm [43] which employs all the features individually in the first step and calculates the misclassification rate using a given classifier. The feature that gives the highest classification accuracy will be selected. Then, all the remaining features will be paired with the first selected feature and using the same approach, second feature is selected. This process continues until the highest classification improvement drops below 3% or misclassification becomes zero. Figure 5.4 shows the flowchart of this process.

5.3.4 Classifiers and Decision Threshold Selection

Once feature extraction and selection tasks are performed, the training and test sets are ready to be fed into the classifier. We use a fusion method combining the results of two

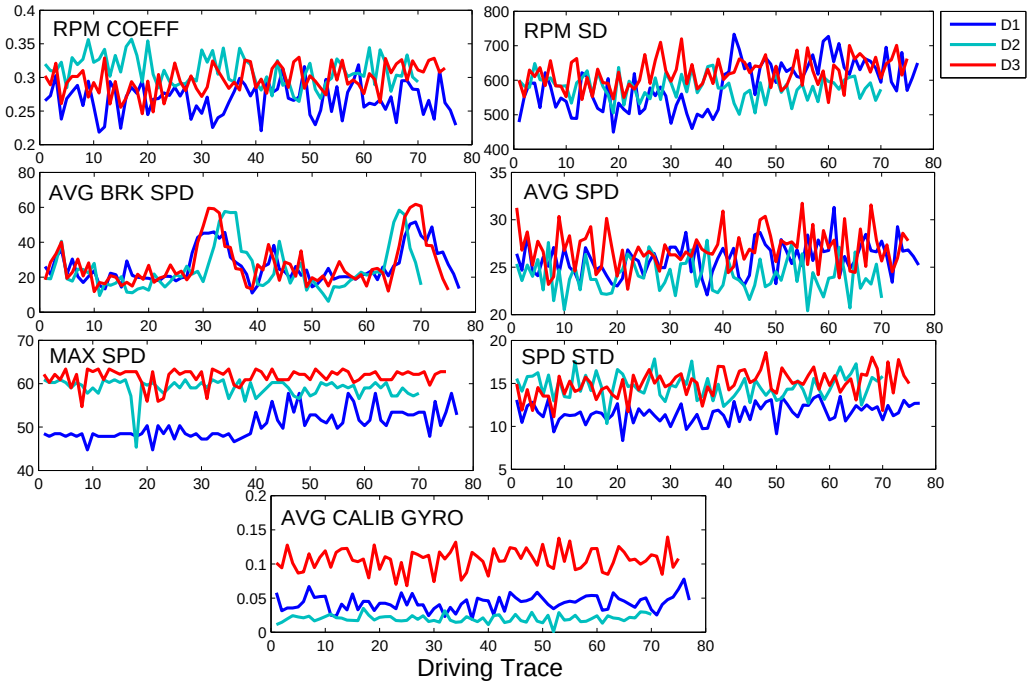


Figure 5.3: Figure shows sequence of all seven extracted features for three participants as a comparison example. The three sequences are more distinguishing in Maximum Speed and Average Calibrated Gyro while they are so similar and close in Average Breaking Speed.

classifiers using an OR logic for final authentication. This means that authorization by either one of the classifier is enough to consider the driver authorized. On the other hand, if both methods reject the driver, then they are unauthorized. The classifiers which are chosen for this purpose are Support Vector Machine (SVM) and Random forest. SVM is selected because it is a strong classifier to deal with complicated recognition tasks as it has several kernel functions suited for variety of purposes. Likewise, Random forest is an ensemble learning method which takes advantage of several weak classifiers and based on voting system gives the final result. Also, SVM is versatile in dealing with limited datasets. The chosen kernel function for SVM is Radial Basis Function (RBF). We used RBF, since our method is non-parametric with complex data and hence complexity of our model should grow based on data. RBF is a squared exponential kernel and therefore complexity of SVM with RBF kernel grows indefinitely. Thus, this kernel is suitable for our approach.

After claiming the identity, the stored driving profile of the driver is retrieved and

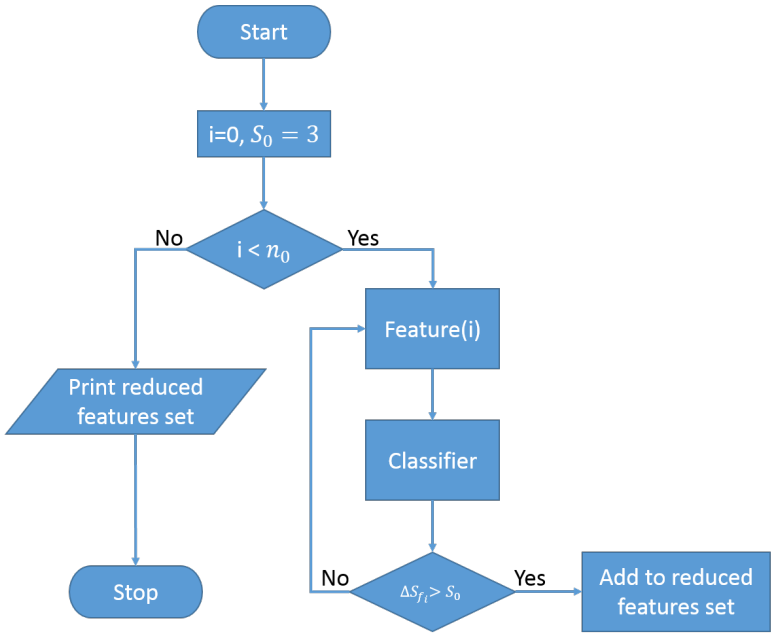


Figure 5.4: The figure shows Greedy Forward algorithm for feature selection. Given a set of features, this algorithm finds the combination of features which gives the highest classification accuracy. In the figure, ΔS_{f_i} represents classification accuracy improvement and S_0 is the threshold to stop the algorithm if ΔS_{f_i} drops below that. Also, n_0 shows total number of features while k indicates number of selected features.

compared against samples derived from their driving session. For template matching and authentication purpose, we use SVM and Random forest in binary classification mode. In this mode, we need to input training data for both classes which are being compared to each other. We assume one of the classes corresponds to one of the driving profiles in the system's database which varies from 1 to 30. The other class which corresponds to the test data, in other words, new driving samples, is always labeled as 0. For example, assume that a person gets into the car and claims that he is driver n, then template of driver n is retrieved which has class label of n. When he starts driving the car, the new driving samples are collected and labeled with 0. The system does not know the real label of the driver and hence labels it with 0 instead. We take 75% of new data and concatenate it with template data to form the training set. Training dataset includes data corresponding to labels n and 0. We input this training set to each classifier and use the remaining 25% of new data for evaluation. In `prediction` part, it is determined that the new data should belong to which of

the classes {0,n} using prediction probabilities. In other words, for each instance of new data, the probabilities that it belongs to the label 0 or n are calculated. Then, based on the prediction probabilities, we select different decision thresholds. Decision threshold is a probability itself which specifies whether a datapoint belongs to class n or 0 based on its prediction probability. If the probabilities for both classes n and 0 are higher than the threshold, it means that datapoint belongs to both classes. If the number of test datapoints which belong to both classes are greater than 50% (majority count), the system concludes that the driver is authorized otherwise if the majority of test datapoints belong to 0 class according to decision threshold, then the driver is unauthorized.

We select the optimum decision threshold (ODT) as the value which gives Equal Error Rate (EER). When the False Acceptance Rate (FAR) and False Rejection Rate (FRR) are equal, that value is said to be EER. We find EER from Receiver Operating Characteristics (ROC) curve as it appears in Figure 5.5. ROC curve visualizes the performance of our fusion classifier. Typically, an ROC curve is expressed as True Accept Rate (TAR), in other words, the complement of FRR, versus FAR. TPR and FRR are defined as below:

$$TAR = \frac{\#of\ TrueAccepts}{\#of\ TrueAccepts + \#of\ FalseRejects} \tag{5.2}$$

$$FAR = \frac{\#of\ FalseAccepts}{\#of\ FalseAccepts + \#of\ TrueRejects} \tag{5.3}$$

In ROC plot given in Figure 5.5, as the curve moves closer to the top-left corner of the box, the better the system is performing. In other words, EER becomes smaller which means better performance. To generate an ROC curve, we calculate (FAR,TAR) points for each decision threshold value and then plot them. Then, if we intersect the ROC curve with $y = -x + 100$ (x and y values in %), we obtain one value which is EER. Also, the decision threshold which provides EER is equal to $ODT = 0.29$.

In some security frameworks, for example in banking systems, a little high FRR for example 5% is acceptable, however, to have 5% FAR is intolerable which means that 5

out of 100 people are falsely authenticated. We use EER in our authentication system, because we want both FAR and FRR be as low as possible.

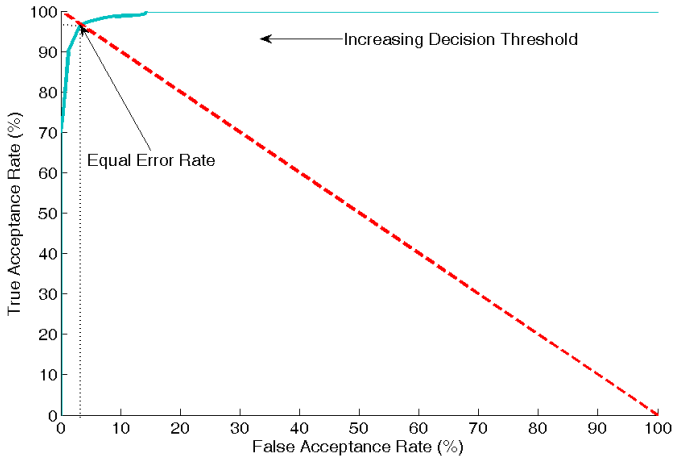


Figure 5.5: This Figure shows ROC curve for the fusion classifier. ROC is a measure of performance for binary classifiers. The more the curve is closer to the top-left corner of the box, the better is the performance. EER is found from ROC curve.

5.4 Summary

We described data processing from the time that we get the driving data to feature extraction and selection. We compared all seven extracted features for three participants as an example and observed distinguishing plots in maximum speed and average calibrated gyro while the similarities were very close in average braking speed. Eventually, we used the selected features for recognition and authentication of the drivers. For authentication, we employed a fusion method combining results of SVM and Random forest classifiers. In addition, we chose EER as a metric for evaluation of authentication performance and explained how we obtained EER from ROC curve.

Chapter 6

Results

6.1 Overview

In this chapter, we present the driver authentication results. First, we analyze the authentication performance for all drivers. Then, we go into individual analysis of drivers and describe their performance in more detail. Next, we explain how authentication performance varies when amount of testing data varies. Finally, we analyze effect of features on driver authentication.

6.2 Implementation run-time

We ran our algorithm for 60 driving data traces belonging to 30 participants. An R implementation of the algorithm processed all traces in less than 100 seconds on a machine with Core i7 2.4 GHz CPU, with an average running time of less than four seconds per person. Implementation in high level languages increases the performance even more by decreasing the run time which suggests that data processing and authentication stages can be accomplished in real time on a smartphone, for example within an Android application. However, data processing begins once enough driving data are recorded during a driving session. Hence, the delay time due to data collection is far more than data processing and authentication altogether.

After training time, drivers need to drive for a time duration in order to become authenticated. It took less than 5 miles of driving to collect enough data for authentication which corresponded to 11.29 minutes for an average speed of 26.06 mph.

6.3 Authentication Performance

Table 6.1 shows the average EER for 30 drivers. A total EER of 4.44% shows our system can authenticate about 95 out of 100 drivers. A smaller EER was obtained for female group by about 1% compared to male group. However, the standard deviation of EER in the female group was higher than male group.

	Mean EER (%)	STD (%)
Males	4.81	4.16
Females	3.89	6
Total	4.44	4.9

Table 6.1: Average EER and STD values. Females have by about 1% less mean EER, however, the standard deviation in their group is higher.

Figure 6.1 shows the histogram of EER values for 30 drivers. From the figure, for 56.67% (17 drivers), the EER values are less than 4%. Also, for 33.33% (10 drivers), EER varies between 6% and 10%. The other 10% (3 drivers) especially the last bar in the figure (2 drivers) show large EER values. This suggests that their odds of becoming either falsely accepted or rejected is high. Furthermore, this could be due to the inconsistency in driving habits. This can also indicate that it is highly possible to authenticate drivers by their driving data if they are consistent.

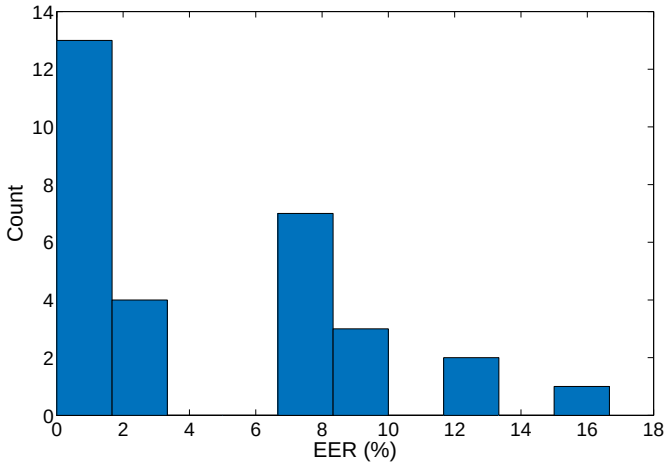


Figure 6.1: Figure shows histogram of EER values for 30 drivers. Each bar represents number of drivers who have the same EER range.

6.4 Individual Analysis of Drivers

Table 6.2 shows analysis of all 30 drivers individually. In the following description, D1 to D30 refer to drivers 1 to 30. For D21, the EER value is the highest (16.67%). This means if they drive the car, there is a 16.67% chance that either they are falsely rejected if they claim to be themselves or falsely accepted if they claim to be others. After that, the highest EER (13.33%) belongs to D16 and D19. As pointed out earlier, this could be due to their highly inconsistent driving habits or unforeseen incidents which happened during the study such as unusual traffic conditions due to road construction. On the other hand, 40% of the drivers had 0 EER value which means that their odds of becoming truly accepted or rejected is high, in other words, with no error. We explain possible reasons for high EER for those participants in following.

First, given surprising differences between EER values, we looked at the data for three participants whose driving traces performed worst when run through the algorithm. We realized that their first driving session was not consistent with their second driving session. Before first driving session, we instructed participants about the route and during the session provided driving guidance well before the intersections. However, having no or little knowledge about the neighborhood and route, they drove more cautiously in the first session compared to second which resulted in noticeable difference of 3 mph in mean Speed between the two sessions.

Second, we note that from Figure 6.1, the two middle bars between 6% and 10% EER are clustered out. We looked at a couple of examples of driving traces from this group. These group received guidance during the first session and drove normally. Furthermore, they said they do not need guidance for the second session as it was exactly the same route. However, they tended to make mistake close to some intersection which were corrected. But, this sudden change of direction had effect on their driving traces.

Therefore, we attribute these poor performances mostly to inconsistent driving traces of those participants than to our authentication method or classifiers.

	Driver#1	Driver#2	Driver#3	Driver#4	Driver#5	Driver#6	Driver#7	Driver#8	Driver#9	Driver#10
EER (%)	0	0	0	6.67	6.67	10	6.67	6.67	0	3.33
	Driver#11	Driver#12	Driver#13	Driver#14	Driver#15	Driver#16	Driver#17	Driver#18	Driver#19	Driver#20
EER (%)	6.67	0	3.33	0	10	13.33	6.67	6.67	13.33	0
	Driver#21	Driver#22	Driver#23	Driver#24	Driver#25	Driver#26	Driver#27	Driver#28	Driver#29	Driver#30
EER (%)	16.67	0	10	0	0	3.33	0	3.33	0	0

Table 6.2: EER values for 30 drivers. The EER for driver n means that all 30 drivers have claimed identity n and their driving data is compared against template driving profile n among which only one is authorized and others are unauthorized.

6.5 Analysis of Features Effect

Figure 6.2 shows how employing different combination of features in driver authentication could affect the EER. In other words, given seven total extracted features, different combination of features are employed to recognize and classify the drivers. First, all seven features individually are used to authenticate drivers. Then, two features out of seven are selected and so on. The total number of combinations is equal to $\sum_{i=1}^{7} \binom{7}{i} = 127$. This analysis shows how the features play role in authentication accuracy.

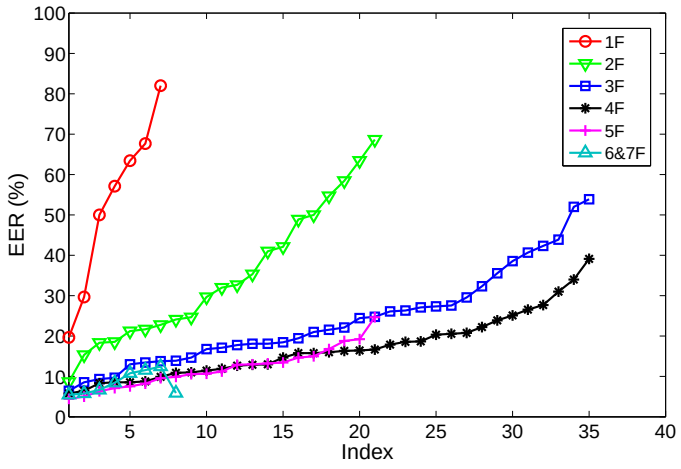


Figure 6.2: The figure shows EER values for different combination of features. (F stands for feature.) This means that each time different numbers of features are employed and authentication is performed and EER is calculated. For example, for 1F plot, all features individually are employed for authentication and EER is obtained, then all EER values are sorted in increasing order. In the horizontal axis, the Index shows the iteration for which EER is calculated. From the figure, as the number of employed features increases, the EER values decrease significantly from 1F to 3F plots but the improvement rate becomes slower afterwards. The lowest EER happens in 5F plot which means that employing 5 specific features results in lowest EER that is 4.44%.

Table 6.3 gives the combination of features which give minimum EER for different number of features. This shows Maximum Speed (Feature 5) is present in all combination of features which result in minimum EER. Also, it is worth to mention that average braking speed (Feature 3) plays no role in the combinations that give minimum EER. Interestingly, when average braking speed is added to 6-features case, the EER increases by 0.45 percentage point, in other words, makes EER 5.89% for 7-feature case.

Table 6.3 also shows Average Calibrated Gyro (Feature 7) improves EER by 11 percentage points from 1 to 2-feature cases. Also, RPM Standard Deviation improves EER by 2.23 percentage points from 2 to 3-feature cases. Features 4 and 1 also contribute positively to authentication accuracy. However, Speed Standard Deviation (Feature 6) increases EER by 1 percentage point which makes the performance worse. Overall, Feature 7 and 4 have the greatest and the least positive impact on authentication performance respectively.

rpm coeff=1, rpm std=2, avg brk spd=3, avg spd=4, max spd=5, spd std=6, avg calib gyro=7		
Number of features	Features which give min EER	EER (%)
1	5	19.67
2	57	8.67
3	572	6.44
4	2457	5.78
5	24571	4.44
6	456721	5.44

Table 6.3: Table shows the combination of features which provides minimum EER for each n-F plot ($n = 1$ *to* 6) as appears in Figure 6.2. Features are shown with their corresponding number.

6.6 Effect of Test Data on Authentication

Figure 6.3 shows that there is a downward trend in EER when test data percentage increases. Here, full test data is 4.9 miles as explained in section 6.2. Starting from 50% to 75%, EER drops sharply and after that slightly decreases until it reaches to its lowest value at 95% of test data. This shows more testing time helps to improve the authentication accuracy. The outcome had been predicted before, since our algorithm

employs classifiers in binary classification mode which requires new driving data (test data) as one of the classes. In other words, it takes some of test data to use in training part along with template data as clarified in section 5.3.4. Therefore, more test data ensures that more data goes for training part and consequently, the accuracy increases.

From Figure 6.3, we also note that slight decrease in EER from 75% of test data onward shows collecting more driving data from some point does not increase performance significantly. In fact, the EER stays the same after 95% of test data onward in our analysis. Therefore, improvement in accuracy of our fusion classifier stops at that point.

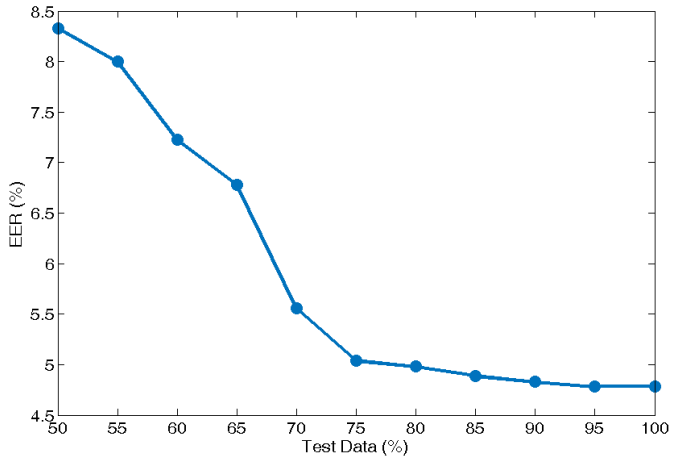


Figure 6.3: The figure shows EER variation when input test data percentage to the system is varied from 50% to 100% of full test data (4.9 miles). EER improves significantly from 50% to 75%, then the improvement rate becomes slower until it stops at 95%.

6.7 Summary

Implementation run-time of our algorithm for each participant is in order of seconds which suggests its real time success in analysis when implemented in the car's computer. We showed driver authentication results on average and individually. Our design showed promising results in total with mean EER of 4.44%. However, the performance was poor for a few drivers which we attributed that to errors in study with reasons. We also analyzed effect of testing data on authentication performance. In other words,

we showed how duration of driving at the time of testing can affect the accuracy of the system. In addition, we described the effect of features on the performance of the system.

Chapter 7

Discussion

7.1 Overview

In this chapter, we discuss about authentication results. We explain how individual features affect authentication performance. Also, we discuss about some methods that we have tried during system design which failed to improve authentication accuracy. Furthermore, we discuss about the failure of our preliminary approaches in our work.

7.2 Factors Affecting Results

We presented a driver authentication system using driving data as an additional security feature for cars. We used Speed, Engine RPM, and Calibrated Gyroscope data to extract features to authenticate drivers. Toward this end, we employed SVM and Random forest, two common classifiers, as the recognizers and combined their results to increase the authentication accuracy.

The system, overall, gives promising results and suggests the feasibility of driver authentication using driving data which can be implemented in modern cars.

We calculated EER values for 30 drivers. For most of the participants, the EER is low enough to be considered good or acceptable. However, for a few of them, high EER suggests some possibilities: 1) inconsistent driving habits during the course of a driving session, for example, sudden acceleration or hard braking in some occasions while doing those actions smoothly in others. 2) blocked roads due to constructions or other issues also affects normal driving pattern and thus it introduces more error.

During the first driving session, we provided driving direction to all participants. We provided these directions well before intersections so that drivers can have ample

time to act. Many participants acknowledged that they do not need direction for the second session. However, we note that some of these participants were corrected when they were about to go a wrong direction. Moreover, we occasionally witnessed unforeseen road obstruction despite conducting study on normal weekdays during the same range of hours. We believe these events have affected the authentication results negatively. We acknowledge that this fact shows three cases of complexities of driving behavior recognition which were mentioned and neglected during system design for the sake of simplicity. These complexities are categorized under *variable driving pattern*, *regional effect*, and *environmental effect*. *Regional effect* and *environmental effect* are more complex than *variable driving pattern* since they are uncontrollable factors by the driver. Considering these three factors is our focus for future research. We also plan to make our system adaptive, in other words, as the drivers' driving behavior change over time, the system adapts itself with their new driving behavior and thus it becomes more robust to habitual changes.

In analyzing effect of features on authentication accuracy, we observed significant improvement in EER when employing 1 to 3 features. However, the improvement rate becomes slow from 3 to 7 features but it is still noticeable. When it comes to number of employed features, we learned that Maximum Speed is the most effective one because it is present in all combinations that give minimum EER. Interestingly, this simple feature plays the most important role. In addition, we note that Average Calibrated Gyro has the greatest impact in improving EER. This feature constitutes total motion of the car around its moving axis, turning maneuvers, and ups and downs of the car, for example, when going over a bump or similar events. Hence, effectiveness of this feature shows: 1) participants act differently on worn out road, in other words, some drive cautiously while some do not; 2) in turning events, participants tend to have different styles. Some take wide turns while others take sharp ones. Also, we learned Average Breaking Speed negatively contributes to the authentication accuracy. This fact could be due to close similarities of this feature for different drivers which is evident from Figure 5.3. We note that the sequence of braking speed mostly includes stopping before intersections or lights and in traffic. However, there might be sudden braking due to unforeseen

incidents as well, for example, an unseen bump or a person jumping into street. But, these incidents never or rarely happened during our study. Excluding those unforeseen events, this shows that most drivers have similar driving habits when it comes to rate of lowering speed before reaching an object or intersection. Thus, our results show Average Speed Braking feature is ineffective in driver authentication.

7.3 Failed Approaches

During system design, we also applied Principal Component Analysis (PCA) to features dataset. However, that did not improve the system's performance and even in some cases, surprisingly made the performance worse. On the other hand, this shows our features are almost non-correlated and thus applying PCA is redundant. Also, this could be result of using PCA with a strong supervised learning algorithm such as SVM. But, possible reasons require further research.

With our preliminary study on spectral analysis of driving data, we learned that this method also could be a potential approach to extract good features to distinguish drivers. Interestingly, we learned that drivers have similar RPM frequency spectrum for their two driving sessions which is different from other drivers. Particularly, we are interested in number of picks and those frequencies at which those occur. However, it was a bit challenging to employ this method in our design since there are more than one dominant frequency in driving data, in this case Engine RPM and also we employed all our features in temporal domain. But, extracting and employing dominant frequencies present in the Speed and Engine RPM data is one of our potential approaches for future work.

We also tried wavelet semblance analysis, finding the phase correlation between speed traces of one driver. We learned this method cannot be applicable to our purpose since it is route dependent and only measures similarities of the two data traces which have the same pattern and shape.

7.4 Summary

We discussed the performance of our design and argued why performance was noticeably poor for some drivers. We attributed these changes to three factors neglected during our system design: *regional effect*, *environmental effect*, and *variable driving pattern*. The latter covers scenarios such as driving habitual changes while the two other factors include situations such as unfamiliarly with the area, unforeseen road obstructions, and adverse weather conditions.

Also, we explained that application of PCA to our features dataset did not improve the results. We discussed effect of features on authentication performance and explained why Average Braking Speed as one of driving features negatively contributes to authentication accuracy. We also discussed about preliminary and failed approaches and proposed some future work to improve the system.

Chapter 8

Conclusions

In this thesis, we introduced DAS, a security platform for cars to authenticate drivers based on their driving behavior. The system stores driving profiles for the first time of driving with the car and then drivers become authenticated after they drive for less than 5 miles given that there exists any driving profile belonging to them in the system's database. We employed seven features extracted from Speed, Engine RPM, and Calibrated Gyroscope data. All these features are extractable from any car equipped with ESC. The first two features could be recorded from ECU and the last from inertial sensors in ESC. The features are designed to authenticate drivers independent of their taken route. We evaluated our system with 30 different drivers who drove the test car and the results were promising with mean EER of 4.44%.

The system performed well overall, however, poor results of some participants suggest that they do not have consistent driving habits when comparing their first and second driving sessions. In other words, they tend to be hesitant in some situations which affects their driving data. Besides habitual changes, other situations contributed to these inconsistencies as well. These situations include: 1) unfamiliarity with the area and 2) unforeseen incidents happened on the road. We categorize all these cases under three factors of variable driving pattern, regional effect, and environmental effect. We note that driver authentication by driving data is a complex problem and therefore we simplified it by neglecting these cases.

We also examined the system for different features and how effective they are in recognizing driving behavior. The results showed that Maximum Speed and Average Calibrated Gyro are the most successful features in driver authentication. Also, we learned Average Braking Speed did not improve authentication results but in fact made

the performance worse.

In addition, we presented the preliminary approaches such as Wavelet Semblance Analysis and DTW and argued that they do not suit our work since they measure similarities of temporal sequences which have same pattern and shape, for example, two speed traces which belong to the same route characteristics. Also, we analyzed RPM data in spectral analysis to find distinct driving characteristics in frequency domain. This approach provided some insight in analyzing driving data in frequency domain, but we described that it is not usable in this work.

To conclude, our work shows that driver authentication based on driving data presents a promising and cost-effective approach in securing contemporary cars.

References

[1] Meinard Müller. *Information retrieval for music and motion*, volume 2. Springer, 2007.

[2] Jim Gorzelany. The most-stolen new and used cars in america. *Forbes*, 2014.

[3] Bureau of Justice Statistics. (2011). *Criminal victimization in the United States, 2008*, Washington, D.C.: U.S. Government Printing Office.

[4] Aki Roberts and Steven Block. Explaining temporary and permanent motor vehicle theft rates in the united states: A crime-specific approach. *Journal of Research in Crime and Delinquency*, 50, 2013.

[5] Federal Bureau of Investigation. (2011). *Crime in the United States, 2011*, Washington, D.C.: U.S. Government Printing Office.

[6] National Highway Traffic Safety Administration (NHTSA)—U.S. Department of Transportation. *Electronic Stability Control (ESC)*, 2011.

[7] Minh Van Ly, Sujitha Martin, and Mohan M Trivedi. Driver classification and driving style recognition using inertial sensors. In *IV*. IEEE, 2013.

[8] Nuria Oliver and Alex P Pentland. Graphical models for driver behavior recognition in a smartcar. In *IV 2000. Proceedings of the IEEE*. IEEE, 2000.

[9] Abdul Wahab, Chai Quek, Chin Keong Tan, and Kazuya Takeda. Driving profile modeling and recognition based on soft computing approach. *Neural Networks, IEEE Transactions on*, 20, 2009.

[10] Xiaoning Meng, Yongsheng Ou, Ka Keung Lee, and Yangsheng Xu. An intelligent vehicle security system based on modeling human driving behaviors. In *ISNN'06*. Springer-Verlag, 2006.

[11] Kristin S. Benli, Remzi Düzagaç, and M. Taner Eskil. Driver recognition using gaussian mixture models and decision fusion techniques. In *ISICA '08*. Springer-Verlag, 2008.

[12] Cuong Tran, Anup Doshi, and Mohan Manubhai Trivedi. Modeling and prediction of driver behavior by foot gesture analysis. *Computer Vision and Image Understanding*, 116, 2012.

[13] C. Miyajima, Y. Nishiwaki, K. Ozawa, T. Wakita, K. Itou, K. Takeda, and F. Itakura. Driver modeling based on driving behavior and its evaluation in driver identification. *Proceedings of the IEEE*, 2007.

[14] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, and Tadayoshi Kohno. Comprehensive experimental analyses of automotive attack surfaces. In *SEC'11*. USENIX Association, 2011.

[15] Christopher Robinson-Mallett. Coordinating security and safety engineering processes in automotive electronics development. In *CISR '14*. ACM, 2014.

[16] Ishtiaq Rouf, Rob Miller, Hossen Mustafa, Travis Taylor, Sangho Oh, Wenyuan Xu, Marco Gruteser, Wade Trappe, and Ivan Seskar. Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study. In *USENIX Security'10*, 2010.

[17] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, and S. Savage. Experimental security analysis of a modern automobile. In *Security and Privacy (SP), 2010 IEEE Symposium on*, 2010.

[18] Aurélien Francillon, Boris Danev, Srdjan Capkun, Srdjan Capkun, and Srdjan Capkun. Relay attacks on passive keyless entry and start systems in modern cars. In *NDSS*, 2011.

[19] Florian Sagstetter, Martin Lukasiewycz, Sebastian Steinhorst, Marko Wolf, Alexandre Bouard, William R. Harris, Somesh Jha, Thomas Peyrin, Axel Poschmann, and Samarjit Chakraborty. Security challenges in automotive hardware/software architecture design. In *DATE '13*. EDA Consortium, 2013.

[20] Lien-Wu Chen, Kun-Ze Syue, and Yu-Chee Tseng. A vehicular surveillance and sensing system for car security and tracking applications. In *IPSN '10*. ACM, 2010.

[21] Zhigang Liu, Anqi Zhang, and Shaojun Li. Vehicle anti-theft tracking system based on internet of things. In *ICVES, 2013 IEEE International Conference on*, 2013.

[22] Christoph Busold, Ahmed Taha, Christian Wachsmann, Alexandra Dmitrienko, Hervé Seudié, Majid Sobhani, and Ahmad-Reza Sadeghi. Smart keys for cybercars: Secure smartphone-based nfc-enabled car immobilizer. In *CODASPY '13*, 2013.

[23] Huaqun Guo, H. S. Cheng, Y. D. Wu, J. J. Ang, F. Tao, A. K. Venkatasubramanian, C. H. Kwek, and L. H. Liow. An automotive security system for anti-theft. In *ICN '09*. IEEE Computer Society, 2009.

[24] Roel Verdult, Flavio D. Garcia, and Josep Balasch. Gone in 360 seconds: Hijacking with hitag2. In *Security'12*. USENIX Association, 2012.

[25] Wael Adi and Nizar Kassab. Hardware architecture for trustable vehicular electronic control units. In *IWCMC '09*. ACM, 2009.

[26] Sakchai Boonmee and Poj Tangamchit. Portable reckless driving detection system. In *ECTI-CON 2009. 6th International Conference on*, volume 1. IEEE, 2009.

[27] Jiangpeng Dai, Jin Teng, Xiaole Bai, Zhaohui Shen, and Dong Xuan. Mobile phone based drunk driving detection. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2010 4th International Conference on-NO PERMISSIONS*. IEEE, 2010.

[28] Derick A Johnson and Mohan M Trivedi. Driving style recognition using a smartphone as a sensor platform. In *ITSC, 2011 14th International IEEE Conference on*, 2011.

[29] Haluk Eren, Semiha Makinist, Erhan Akin, and Alper Yilmaz. Estimating driving behavior by a smartphone. In *IV, 2012 IEEE*, 2012.

[30] Johannes Paefgen, Flavius Kehr, Yudan Zhai, and Florian Michahelles. Driving behavior analysis with smartphones: insights from a controlled field study. In *Proceedings of the 11th International Conference on mobile and ubiquitous multimedia*. ACM, 2012.

[31] Dante Papada and KW Jablokow. Conceptual design of a driving habit recognition framework. In *CIVTS, 2011 IEEE Symposium on*. IEEE, 2011.

[32] Wen-Chih Hsiao, Mong-Fong Horng, Yun-Je Tsai, Tsong-Yi Chen, and Bin-Yih Liao. A driving behavior detection based on a zigbee network for moving vehicles. In *TAAI, 2012 Conference on*, 2012.

[33] Jin-Hyuk Hong, Ben Margines, and Anind K. Dey. A smartphone-based sensing platform to model aggressive driving behaviors. In *CHI '14*. ACM, 2014.

[34] Sicco Verwer, Mathijs De Weerdt, and Cees Witteveen. Learning driving behavior by timed syntactic pattern recognition. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Two*. AAAI Press, 2011.

[35] Wathanyoo Khaisongkram, Pongsathorn Raksincharoensak, Masamichi Shimosaka, Taketoshi Mori, Tomomasa Sato, and Masao Nagai. Automobile driving behavior recognition using boosting sequential labeling method for adaptive driver assistance systems. In *KI '08*. Springer-Verlag, 2008.

[36] Xiaoning Meng, Ka Keung Lee, and Yangsheng Xu. Human driving behavior recognition based on hidden markov models. In *ROBIO'06. IEEE International Conference on*. IEEE, 2006.

[37] Toshihiro Wakita, Koji Ozawa, Chiyomi Miyajima, Kei Igarashi, Katunobu Itou, Kazuya Takeda, and Fumitada Itakura. Driver identification using driving behavior signals. *IEICE - Trans. Inf. Syst.*, E89-D.

[38] The obd-ii home page. `www.obdii.com`. Accessed: 2015-03-16.

[39] A. van Zanten. Bosch esp systems: 5 years of experience. *SAE Technical Paper*, 2000.

[40] GRJ Cooper and DR Cowan. Comparing time series using wavelet-based semblance analysis. *Computers & Geosciences*, 34, 2008.

[41] Donald B Rubin. *Multiple imputation for nonresponse in surveys*, volume 81. John Wiley & Sons, 2004.

[42] Donald B Rubin. Statistical disclosure limitation. *Journal of official Statistics*, 9(2):461–468, 1993.

[43] B.D. Fulcher and N.S. Jones. Highly comparative feature-based time-series classification. *Knowledge and Data Engineering, IEEE Transactions on*, 26(12):3026–3037, 2014.