Smartphone Application to Detect Texting While Driving

Detecting Texting while Driving

Design Document

Group:
sdmay19-16
"The Face Mates"

Client:
Daji Qiao

Faculty Advisor:
Daji Qiao

Team Members:
Ryan Baker - Lead Architect
Derek Clayton - Report Manager
Lucas Golinghorst - Test Engineer
Andrew Knaack - Lead Designer
Sara Mace - Meeting Scribe
Kristina Robinson - Project Lead

Team Website:
sdmay19-16.sd.ece.iastate.edu

Version:
1.0

Version Date:
10/12/2018

Table of Contents

List of Figures

List of Tables

List of Symbols

List of Definitions

- False positive: In this case, a false positive is our application incorrectly identifying a passenger as a driver and blocking their texting capabilities.
- False negative: In this case, a false positive is our application incorrectly identifying a driver as a passenger and not blocking their texting capabilities.
- TwD: An acronym which stands for Texting while Driving

1. INTRODUCTION

1.1 Acknowledgement

We want to acknowledge our client and advisor, Daji Qiao, for thinking of this project. We also want to acknowledge the department of Electrical and Computer Engineering at Iowa State University for encouraging us to work on professional projects and providing us with extra phones for development purposes.

1.2 Problem and Project Statement

Our project will address the issue of texting while driving. According to teensafe.com, just 3 years ago, 391,000 injuries were caused by distracted driving accidents. Even more serious than that, 9 people are killed everyday from distracted driving accidents. Answering texts while driving might seem harmless but replying to a text message takes the driver's eyes off of the road for at least a few seconds. It takes only three seconds of the driver having their eyes off of the road to be in a car crash. While texting and driving is not the only cause of distracted driving, approximately 660,00 drivers use a cell phone during the daytime which means that the chance for a TwD accident is significantly increased every day. In order to find a solution to this problem, we will build an android application to detect if someone is texting while driving. Accurately detecting whether someone is in the driver's seat and that they are texting while driving is our biggest challenge. In order to do that without simply locking out everyone from their phones, our solution will have to include many different measures to ensure accurate detection.

The solution to this problem will be to develop an android application to detect whether someone is TwD. Once TwD has been detected, the user will not be able to send text messages until they stop driving. We will ensure that we can accurately detect TwD by checking multiple different categories. The first thing we will check is if the driving is moving faster than 10 miles per hour. This will tell us if they are in a moving vehicle. The second category we will look at is determining where the person is sitting in the car. We will use cameras to determining what is in front of the person as well as seeing which way the seatbelt is going across the users chest. Our last category of verification is learning how the user normally uses their phone so we will be able to find differences when they are TwD.

1.3 Operational Environment

The operating environment will be the user's vehicle. We expect that this environment will not be subject to any extreme conditions, but rather a controlled environment. If an extreme condition does occur (such as a car accident), then the app is no longer of use in that environment anyway. It is also worth noting that the application is intended to be active even when the user is not in a vehicle, since it is vital for the program to understand a user's regular texting habits when not driving. Our application will be downloaded onto the user's phone, so withstanding hazards in the environment it is used in is not much of a concern.

1.4 Intended Users and Uses

Our intended users are anyone who drives a vehicle and has an Android smartphone. As our app is meant to detect texting while driving, if someone is not a driver then it doesn't make sense for them to have an app that prevents them from texting and driving. Our user also needs to have an android smartphone as the app we are making is an android app, so therefore they need an android phone to run the app.

The intended use of our app is for anyone who wants to make sure that they cannot text and drive. For example, this could be a parent who has their child install it because they want to make sure the child stays safe; or, someone who acknowledges that they have a problem with texting and driving and want to take measures against their habit.

1.5 Assumptions and Limitations

Our list of assumptions is as follows:

1. The phone used during driving is being consistently used by the driver. They will not use someone else's phone while in the driver's seat. This assumption needs to be made since we will use machine learning to track the phone owner's tendencies. If the driver uses someone else's phone then machine learning will not help us with detection.
2. The user has the app enabled consistently, even when not driving. This will allow us to learn the owner of the cell phone's texting tendencies when they are not driving so we can more accurately detect texting while driving.
3. The app should be relatively simple to activate. Therefore, the user does not have to go through many steps themselves to make it functional. The user will be unwilling to use the application if it is too complicated to activate.

4. The driver, if texting, will only ever be texting with one hand. This is the assumed normal behavior. Detecting texting while driving will is simplified by this assumption if we can figure out a way to determine how many hands they are using to type on their phone.
5. Our application will only be used by drivers in Iowa. Only letting our application be used in Iowa will simplify the data privacy laws that we need to follow.
6. People who are running will not be texting. Since runners can reach very fast speeds that are over our speed threshold, we will assume that the user will not be using their phone if they are running as that is difficult to do.

The project limitation are as follows:

1. The application must not take up unreasonable amounts of memory or battery life to run. This could cause problems or irritation for the user if violated.
2. The application will support English and not any additional languages or translations.
3. The application will not make use of any external hardware. All of the sensors utilized will be on the phone.

1.6 Expected End Product and Deliverables

The end product is a standalone Android application which detects texting while driving. This will be done with machine learning techniques and readings from the phone sensors. This application will not require any additional hardware to be run.

Since this project is split across two semesters, there will be a preliminary end product by the end of the first semester. This will be a working prototype with all intended functions of the final product in use while the application is running, but not necessarily perfected or well-developed by that point.

The application will be capable of performing multiple tests to determine the user's position in the vehicle. Such tests will include speedometer tests, vertical/centripetal acceleration tests, GPS velocity tests, texting speed/typo tests, phone handling tests, and phone camera tests. Using the data collected from these real-time tests, the application will determine whether the user is driving or not.

2. Specifications and Analysis

2.1 Proposed Design

There following are six technologies which make up this project.

Speedometer: The first and most fundamental is the speedometer, a sensor Android phones have built-in. This will determine if the system is trying to detect texting while driving at all. If the speedometer is below a safe speed, the system will switch to its learning mode. If it is above safe speed, the system will switch into detection mode and begin calculating the probability of texting while driving.

Texting Speed: When not in detection mode, key input will be captured while the texting application is in use so that the system can learn the user's typical texting speed. During detection mode, the system will take this norm and compare it to how fast the user is currently typing. If the texting speed is significantly slower, it implies the user may be distracted (possibly driving).

Spell-Checking: When not in detection mode, the user's captured key inputs will be analyzed for spelling errors so that the system learns how many errors are in the average sentence the user types. During detection mode, the system will take this norm and compare to the number of errors in what the user is currently typing. If the ratio of misspelled words is significantly higher, it implies the user may be distracted (possibly driving).

Centripetal Acceleration: Using GPS functionality, it can be determined what side of the vehicle the user is in when it makes turns. If the GPS determines the user is one the right side of the vehicle, then the user cannot be in the driver's seat and detection mode will be disabled. If the car encounters a bump in the road, GPS functionality can also determine if the user is in the front or back of the vehicle. If GPS determines the user is in the back, then detection mode will also be disabled. If the user is determined to likely be the driver, spelling and texting speed indications are much more likely to disable the phone's texting functions.

Phone Camera: While in detection mode, the front and back cameras of the phone can be used by the system to try to figure out where the phone is in the car. If the system can identify key features of the car like the wheel or the seatbelt, it can provide information about where the user is likely sitting (i.e. the direction of the seatbelt across the user's body can tell which side of the car the user is in).

Phone Handling: While in learning mode, the system will try to become accustomed to the user's phone-handling habit, such as the user's dominant hand, how often they use the phone with one hand, etc. In detection mode, this information will be used to detect if the user is handling the phone significantly differently than normal. For example, if the user almost always texts with two hands, but is trying to text with only one hand while the system is already in detection mode, it is more likely that the other hand is being occupied steering the wheel (indicating texting while driving).

The combination of these technologies should be able to work together to consistently learn when users are texting while driving and when they are not.

2.2 Design Analysis

Up to this point in the project, we have done many weeks of research to come up with a solution that is unique and effective. The research we have done includes technologies that have been tried previously, the effectiveness of current research, and implementation of possible solutions. Throughout our research we came up with thirteen possible technologies to use in our solution. The following are the thirteen technologies we researched.

1. Analyzing texting speed when user isn't driving compared to when they are driving.
2. Analyzing grammatical errors in texts when the user isn't driving compared the when they are driving.
3. Using NFC tags in the steering wheel and drivers seat to detect the driver's smartphone. Being detected indicates that the user is definitely the driver.
4. Using the phone's built-in speedometer to determine if there is a possibility that the user is in a fast-moving vehicle.
5. Using a camera on the dashboard to track eye movement and facial positioning to see if the user has their eyes off the road.
6. Using centripetal acceleration and angular speed via multiple technologies to determine if the smartphone is on the right side or the left side of the car.
7. Using vertical acceleration when going over bumps via the same technologies to tell if the smartphone is in the front of the car or the back of the car.
8. Using the front facing camera in a smartphone to determine which side of the car the user is sitting on based on the way the seatbelt is laying across their chest.

9. Using the front facing camera in a smartphone to find predictable constants in a car.
10. Using the back camera to find either the steering wheel, the glove box, or the back of a seat to tell us whether the user is in the front or the back of the car. If they are in the front, the images would be used to determine if they are in the passenger's seat or the driver's seat.
11. Monitoring leg lift with a smartphone in the pocket to tell us which side of the car the user entered on.
12. Using the speakers in a car to send frequencies to the smartphone to determine the location of it in the car.
13. Analyzing the way a user handles their smartphone when they are not driving compared the when they are.

Once we came up with the thirteen technologies we might want to use, we met as a team to talk about the feasibility of each solution. The factors we took into consideration to determine if a component was feasible include: how complex the approach is, hardware requirements, resource cost, and the availability of data to test each approach. Based on these factors, the six possible components we chose to test for our solution include the six technologies discussed in the proposed solution section above: speedometer, texting speed, spell-checking, phone cameras, centripetal acceleration, and phone handling. Now we are in the process of testing each technology to see if they will be effective solutions.

Based on the research the team has done, these six technologies seem to be the most promising for giving us useful data, as well as being able to implement them in the time frame we have to work on the project. The first three technologies (analyzing velocity, texting speed, and spelling) are the simplest and most easily-accessible out of all six. Using the speedometer is the most feasible and the first part of the solution that has to be implemented. Determining if a phone is in a moving car will be important for deciding whether our application should be in learning mode or in detection mode. Analyzing texting speed and looking at grammatical errors will be used to learn the user's texting habits. We will then be able to use this data to detect when their texting speed or amount of grammatical errors differs from the users normal habits significantly enough to suggest they may be TwD.

The last three technologies (analyzing cameras, centripetal acceleration, and phone handling) we have proposed either do not have much research to learn from or are new ideas that we came up in collaboration with our client. Here we will discuss how we want to use these components and the challenges we will face with their implementation.

Using GPS to determine the centripetal acceleration on the phone can tell us whether the user is on the left or right side of the car when the driver is making turns. To determine if the user is in the front of the car, we can use GPS when the car is going over bumps in the road because the upward acceleration experienced by the phone is noticeably different between parts of the car. This technology will be key to helping us determine what seat the user is in. The challenges we face with this technology are that there is very little research on this approach and we would have to assume that the driver is going to have to make a few turns and drive over bumps so we can determine where they are positioned in the car. The next technology is using the front facing camera to find predictable points in a car. By predictable points we are referring to points in cars that would be common throughout different models and sizes of cars. An example of a predictable point would be the windows on each of the car doors. This is a new proposed approach from our client. We will need to collect data from many different kinds of cars to see if we can use this approach to help us determine if we can accurately and consistently locate where the user is in the car. The last technology we are looking into is also a new approach that would learn how a user normally holds their phone so we can find inconsistencies in the way they hold their phone when they could potentially be driving. Since this is also a new idea, we will need to figure out what kind of data will be useful to collect and how we can use that to learn about how users handle their phone.

3. Testing and Implementation

3.1 Interface Specifications

We will use a computer to check the functionality of tests and components of our application. Customizable unit test cases will allow us to input values and see the result in the application. The speedometer on the car will also serve as a means of double-checking the readings on the application.

3.2 Hardware and software

The hardware that we will be utilize to test is the phones provided by the advisor and any personal devices that we want to use to test the app. We will also be using a car or bus when we are testing the aspects that need to detect moving above ten miles per hour in the car. Even though we won't be using any physical parts of the car, the car itself is needed to test the app. We could also possibly be using some sort of hardware to distract the user to a degree that it would be similar to texting while driving.

3.3 Functional Testing

System testing

We will need to test each phone we have to make sure that the hardware can support what we want even if the software does and it theoretically should. We are currently unsure of all the tests, but making sure the GPS functions work is something we will definitely do. We will also do hardware testing with multiple phones simultaneously, making sure that under the same conditions, similar results should happen. We will also do testing to see the false positive and negative rates, and from that seeing how we can lower those.

Unit Testing

We most likely will have 2-4 cases that test whether a person is in a vehicle, and if they are, if they are driving or if they are a passenger. We plan to test all of these individually as unit tests before integration testing will be done. The units will most likely be broken up further in the future. Most of this testing will occur on a moving vehicle with the user as the passenger, or simulating the driver by distracting the tester to a similar degree.

Acceptance testing

We will base our acceptance tests off of reaching each of our five milestones. These milestones are accelerometer test to enable mode switching, completed prototype of learning mode, completed prototype of detection mode, making enhanced versions of learning and detection mode based off of results from the prototypes, and the last milestone which will be guaranteeing false positive and false negative rates somewhere at or below 10% for both false positive and false negative.

3.4 Non-Functional Testing

One way we plan on testing performance is measuring the percentage rate that the app crashes during testing. We also plan on measuring the number of times the app correctly detects TwD. This number will be shown as a percentage so we can identify how well the app is performing at its job. For compatibility testing, we plan on testing the app on multiple brands of android phones. We also plan on testing on the different versions of android. We will make sure the app is secure to make sure the user's data is not being lost when they delete the app and redownload the app.

3.5 Process

Speedometer: This is the simplest of tests that will do. We will use multiple phones that our application is running on. We will put all these phones in the same vehicle and will see if they detect if the vehicle is going above the allowed speed. This should be a 100% detection rate so we will know we have done something wrong if this doesn't work as intended.

Texting Speed: For this test, we plan on measuring how fast the user normally texts, when they are not driving. We will then safely distract a user of the application while they are in a moving vehicle to then measure how fast the user texts when distracted.

Spell-Checking: This test will measure how how many errors a user typically has when texting when not driving. We will then compare how many errors a user has when they are safely being distracted in a moving vehicle. To make sure these tests are accurate we will have multiple users do the tests and use multiple phones to see if there is a difference.

Centripetal Acceleration: This test will use centripetal acceleration to determine the location of the user in the car. It will use GPS to detect turning and combine this data with known velocities, taking data on left and right turns. It will then determine the the average of these turns combine to find the central reference point of acceleration. If the phone's acceleration is less than or greater than the acceleration of the reference point, it indicates that the user is on one side of the car. We will safely test this by having a passenger user use the application. If the application can detect if the user is a passenger, it should be able to detect if a user is in the driver's seat.

Phone Camera: Test images will be taken for comparison of images captured while user is in the car. Tests will be done to determine accuracy of detecting when user is in the car from these image captures.

Phone Handling: This test will use a machine learning algorithm to learn how the user typically holds their phone under different conditions. From data collected, the application will discover how the user holds their phone in the event that they text and drive. If the handling that the machine learning algorithm learns, confirms a texting while driving handle on top of positives from other tests, then it will detect texting while driving.

The figure below represents the order in which these technologies must be implemented and working. The speedometer is most fundamental, since it determines which technologies are active, and if they are in learning mode or detection mode. The technologies which are active during both detection mode and learning mode must have a functioning learning mode first, since detection requires information gathered in learning mode to do anything. The technologies to the left of the Speedometer determine positioning in the car, and those on the right determine the probability of TwD.

At least one technology to detect TwD and one for user position must be functional before the system can be allowed to lock out the user. If only one half of the functionality is present, passengers will either be accidentally locked out of their phones, or a driver who *is* TwD will not be.

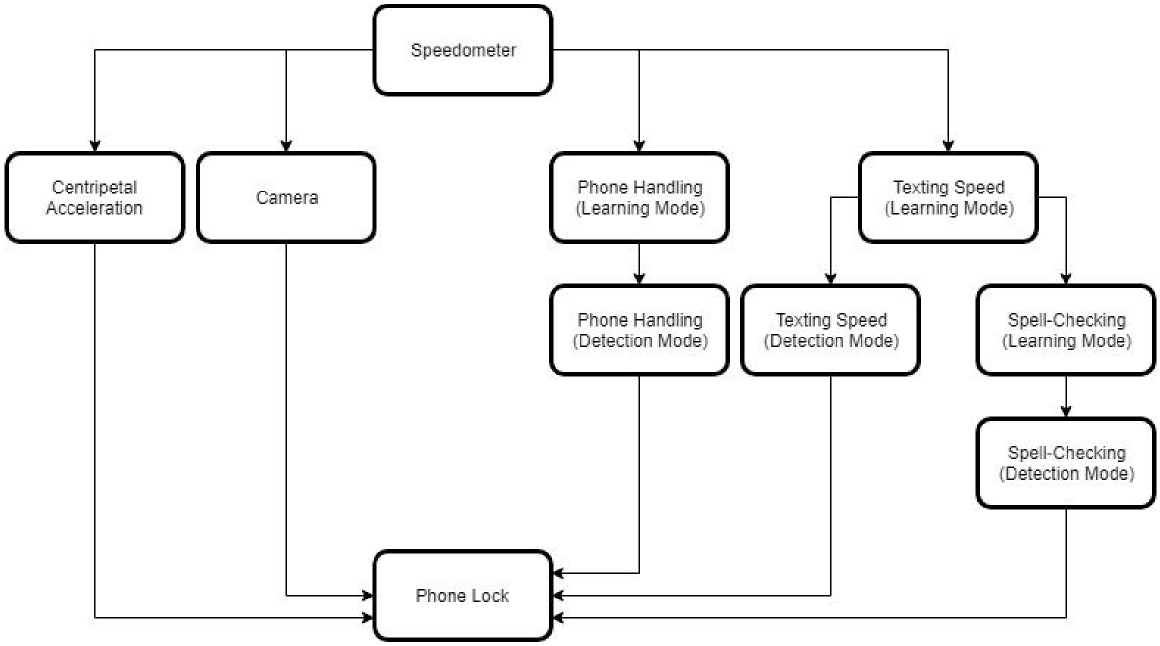

Figure 1: Process Flowchart

3.6 Results

We have not had any testing done so far, most of what we have done is research into determining what may and may not work. The next step in our project is to start collecting data for each of the six proposed technologies to determine feasibility. Once data collection is done, we can start developing and testing each component of the application.

4. Closing Material

4.1 Conclusion

Our project aims to provide a solution that can reliably detect texting while driving. To accomplish this, we will develop a standalone android application using android developer software. Each week we will iteratively develop a more functional version of the solution and present to the client for feedback. This project will be heavy on research and prototype testing as we experiment the feasibility and accuracy of different solutions. A number of solutions of varying complexity will be explored. Our ultimate goal is to create an application that will utilize the sensors embedded in the phone to successfully detect texting while driving. Once detected, a protection system will activate that makes the user unable to send text messages until they stop driving. The delivery of this solution will provide a means for safer driving through text prevention.

4.2 References

"100 Distracted Driving Facts & Statistics for 2018." *TeenSafe*, 5 July 2018, www.teensafe.com/distracted-driving/100-distracted-driving-facts-and-statistics-2018/