CAReful

Giuseppe Cancello Tortora, Mirko Casini,
Andrea Lagna, Martina Marino

g.cancellotortora@studenti.unipi.it, m.casini16@studenti.unipi.it
a.lagna1@studenti.unipi.it, m.marino29@studenti.unipi.it

Abstract

Around the world, road accidents cause 1.35 million deaths a year, and have now become the leading cause of death between the ages of 5 and 29 [1]. CAReful is an application that can be used while driving to detect dangerous behaviours, score them and alert the driver in case of exceeding a certain threshold. It uses different sensors to monitor the driver, such as microphone, GPS, camera, accelerometer, gyroscope and magnetometer.

1 Introduction

Distracted driving is any activity that diverts attention from driving, including talking or texting on your phone, eating and drinking, talking to people in your vehicle, fiddling with the stereo, entertainment or navigation system or anything that takes your attention away from the task of safe driving.

Texting is the most alarming distraction. Sending or reading a text takes your eyes off the road for about 5 seconds. At 55 mph, that is like driving the length of an entire football field with your eyes closed. You cannot drive safely unless the task of driving has your full attention. Any non-driving activity you engage in is a potential distraction and increases your risk of crashing. Using a smartphone while driving creates enormous potential for deaths and injuries on roads. According to NHTSA [3], in 2020, 3,142 people were killed in motor vehicle crashes involving distracted drivers.

2 Goal

The goal of this project is to develop an Android application for smartphones able to measure the level of disattention while driving. The app makes use of different sensors provided by the smartphone in order to collect information about the behaviour of the driver while driving and of the environment around him (e.g., tortuosity of the road can have an impact on the attention level during the trip).

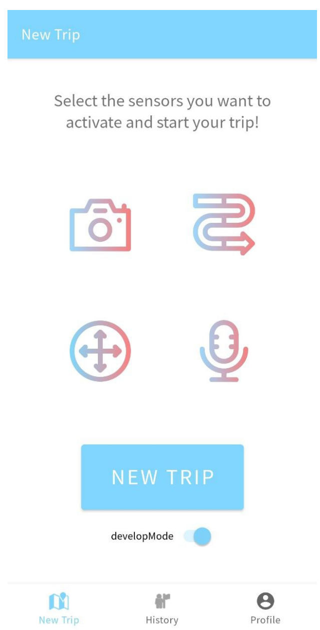

Figure 1: NewTrip screenshot

3 Sensors

All the sensors used are briefly described in the next paragraphs and a quick introduction on their usage in CAReful is done. The main sensors useful for the proper functioning are: GPS, microphone, camera and all the motion sensors such as accelerometer, gyroscope and magnetometer.

The develop mode is implemented in order to make the user understand the functioning of individual sensors. It can be activated using the appropriate switch in the new trip page, then clicking on the icon of the sensor that is going to be tested to see the specific functioning.

3.1 GPS

GPS is a mandatory sensor used to start a trip and register all the information at the end of it. Every trip has departure and arrival coordinates and an on-the-fly updated map is shown during it. With this sensor is also possible to retrieve the speed of the vehicle and evaluate an average value that will be used with the curvature index as a multiplying factor in the score formula.

In order to use Google API for showing the interactive map, a project on Google Maps Platform [4] has been created to obtain the credentials needed. The map is updated on the current location every 5 seconds.

3.2 Microphone

The microphone of the smartphone is used to capture the noise in the car during the trip. The android.media AudioRecord class is used to sample the audio every 500ms which will be saved into a buffer. For each element of the buffer it is computed the sum of the squares, that is then divided by the size of the buffer in order to obtain the mean value. This value is then used to calculate the decibel measure as:

$$decibelMeasure = 10 * \log_{10}(mean)$$

This value is then compared with a threshold of 150.0 dB to evaluate if the volume exceeds it: in case of threshold exceeded for 5 consecutive samples, an alert signal occurs. In develop mode, it is also possible to see how the sensor works by checking the decibel text field that is updated live and the alert that is raised once the noise is detected.

3.3 Camera

The camera sensor is used for the detection of the driver drowsiness and to recognize if the driver has turned its head to the left or to the right, losing the focus on the road. The Google ML Kit was exploited and in particular the Face Detection APIs.

The camera preview FPS are set to 10 due to a trade-off between the performance in the detection and the battery consumption, and for obvious reasons the front camera is exploited. The settings LANDMARK MODE ALL and CLASSIFICATION MODE ALL are used to detect the points of interest within a face and the probabilities of the eyes to be open.

For what concerns the driver drowsiness, the two values of left and right eye open probabilities are considered with a threshold of 0.5. Once the driver has "closed" his eyes for 20 consecutive frames (∼2 seconds), the drowsiness counter is incremented.

For what concerns the head orientation, the euler Y angle is used with a threshold of 40 degrees to the left and to the right. Once the driver has turned its head in one of the two directions for at least 4 seconds (40 consecutive frames), the head counter is increased.

In develop mode it is possible to see a bounding box around the face detected with a live update of all the values needed for the detection of the drowsiness and head orientation.

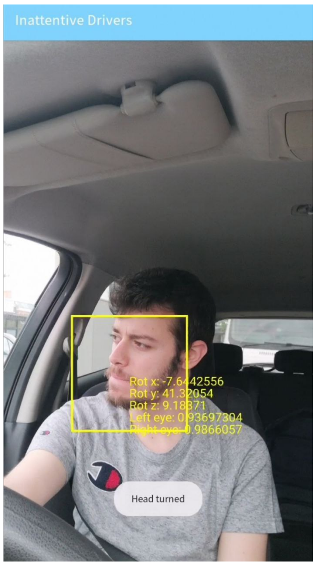

Figure 2: Head turned example

3.4 Accelerometer

The accelerometer sensor has a dual purpose: it is fundamental in fall detection and is also used to detect and compute the road curvature by working together with the magnetometer. For what concerns the fall detection, the accelerometer is responsible for the computation of acceleration's magnitude. The magnitude is computed using the following formula:

$$magnitude = \sqrt{x^2 + y^2 + z^2}$$

then is stored in a small array that contains always the last 8 magnitudes computed.

The listener of the accelerometer is activated at regular intervals with a value of SENSOR DELAY NORMAL (∼200 ms) and is evaluated using two values of threshold: lowThreshold and highThreshold that are respectively 5.0 and 18.0. According to [2], every time the algorithm searches in the array for a value lower than lowThreshold and, if found, it searches for a value higher than highThreshold in the remaining part to detect a fall.

In develop mode a text field is updated with the new value of the counter once a fall is detected.

The accelerometer is also used in the road curvature combined with the work of the magnetometer as described in the next paragraph.

3.5 Gyroscope

The gyroscope sensor is used to detect if the phone is used by the driver during the trip. The usage of the values obtained from the gyroscope is extremely easy but at the same time extremely effective. It is sufficient to detect and store the value measured from the gyroscope and check if one of the three axes value is above a certain threshold (set to 1.9f). Obviously it is necessary to evaluate rotations in one of the three axes, both clockwise and counterclockwise, therefore the threshold is evaluated in module. The listener of the gyroscope is activated at regular intervals with a value of SENSOR DELAY NORMAL (∼200 ms). In develop mode a text field is updated with the new value of the counter once a the usage is detected.

3.6 Magnetometer

The magnetometer works alongside the accelerometer in order to compute the curvature index of the road covered during the trip. The first thing to do is to compute the azimut simply by taking the value obtained from the magnetometer, then all the values (sampled each 200ms) are added up and, using two different thresholds, the curvature index of the road is computed.

Some clarifications are necessary, first of all the values obtained from the magnetometer are passed through module operator due to the fact that for us the sign is not relevant, in addition to this is necessary to specify that the value of the threshold are set to 10 and 60. This choice is driven by some test data that led us to classify the road in three different categories: easy road with a curvature index of 1 (value of the sum lower than 10), medium road with a curvature index of 2 (value of the sum between 10 and 60) and hard road with curvature index of 3 (value of the sum higher than 60). In the Section 4 the usage of this value will be explained better.

In develop mode a text field is updated live with the two values of the sum and the curvature index.

4 Score

In this section the formula used to compute the score of the driver is explained. All the values are normalized in the range 0-1 by dividing the counters of the detections by the maximum values obtainable in a minute. This is due to the fact that every minute the attention level is re-computed (and the counters reinitialized) to have a better accuracy in the final formula. The left part of the formula developed is a combination of the average speed of the vehicle and the curvature index of the road, with the purpose of weighing the values coming from the other sensors which are strictly related to the driver disattention (right part of the formula).

$$disattentionLevel = (\frac{speed}{4} + \frac{curvature}{3}) \cdot (\frac{head}{12} + \frac{drowsiness}{30} + \frac{noise}{12} + \frac{fall}{5} + usage)$$

The formula below is used to compute the driver's disattention level at certain time, then the previous disattention level is weighted by the alpha parameter (set to 0.7) and added to the new disattention level according to the following formula:

$$disattentionLevel = (\alpha \cdot disattentionLevel) + (1 - \alpha) \cdot prevDisattentionLevel$$

The opposite of this score is finally used in order to obtain the driver's attention level that will be displayed during the trip and saved in the history trips of the driver.

$$attentionLevel = 1 - disattentionLevel$$

5 Conclusion

The application is extremely useful and is designed for different uses: starting from a common driver that wants to check if his driving style is correct and he is not distracted at driving, but also for a company that manages buses and wants to know if the bus drivers are good at their job and do not jeopardize the safety of bus passengers. Obviously there are a lot of possible improvements that can be developed in order to increase the efficiency of the application: some possible examples are introducing a specialized audio recognition system that recognizes voice or music in such a way as to weigh them differently, adding a classifier able to recognize if the driver has something in his hand and maybe even say what and using a statistical method that calculates the degree of difficulty of the road travelled and uses it as a multiplication factor in the inattention index.

References

[1] Redazione ANSA (2018) *Nel mondo 1,35 milioni di morti per incidenti stradali*, www.ansa.it

[2] Tran Tri Dang, Hai Truong, Tran Khanh Dang (2016) *Automatic Fall Detection using Smartphone Acceleration Sensor*, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 12.

[3] NHTSA (2020) *Distracted Driving*, www.nhtsa.gov

[4] Google (2022) *Google Cloud Platform*, console.cloud.google.com