Hugh Ashley, David Singh, Zain Cruz

Group Evo Squad

EEL5632 Project

"A Driver-Performance Based Approach to Vehicle Safety and Warning Systems using Fuzzy Logic"

Abstract

With the rise of ADAS in modern vehicles, the role of the driver is being slowly diminished. We conversely design a simple system to instead reward attentive, controlled driving rather than encouraging the opposite. Our system determines a driver's score based on various inputs reached via the CAN bus that then penalizes or rewards the driver based on safe habits. We note the immense area to expand via other sensors, vehicular adaptability, and configurability.

Introduction and Motivation

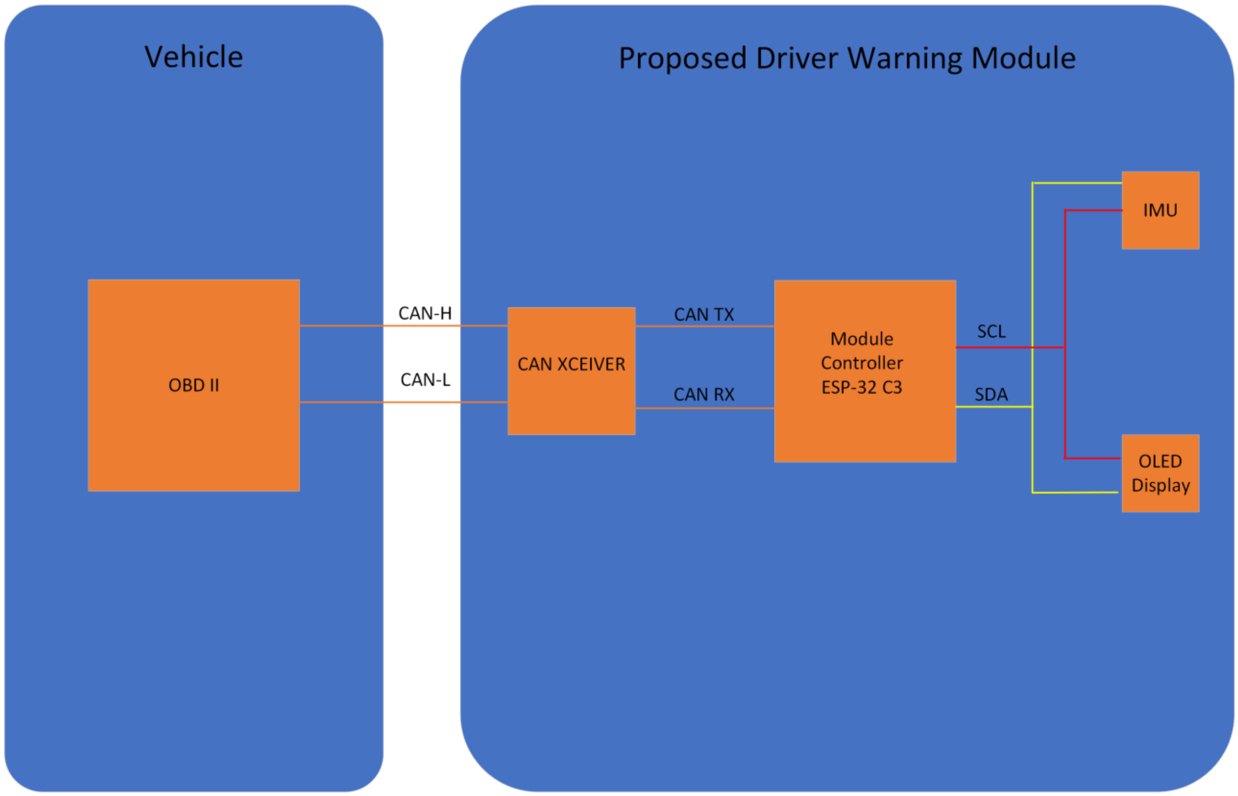
We are currently in the middle of a boom in ADAS – Advanced Driver Assistance Systems, as the automotive industry transitions in the effort to make personal and commercial vehicles autonomous, or self-driving. In this transitory period, most new vehicles on the road active systems, like adaptive cruise control, lane-keeping/centering assist, and automatic emergency braking. This is in addition to passive systems like blind-spot monitoring and rear cross-traffic alerting. Despite the goal of ADAS being to improve road safety and partially offload the driver, ADAS has been encouraging distracted driving and its subsequent consequences [1].

In contrast to ADAS, our system rewards driver attention and punishes distraction and poor driving style. We have proposed, developed, and tested a system to do just this based on vehicular sensors available on the CAN bus (accessed through OBD2 diagnostic port) and an IMU (Inertial Measurement Unit). We then compute a driver score to indicate a generalized report of the user's driving habits as it pertains to smooth, safe inputs. The driver is subject to varying amounts of warning mechanisms based on the driver score, updated in real-time. These warnings become increasingly irritating to the driver as the driver score declines, while dissipating as the driver score raises. More details will be given in the methodology section.

Methodology and Experiment

The project hardware consists of a CAN transceiver connected to an ESP-32 microcontroller. It should be noted that the specific microcontroller is mostly irrelevant – the primary dependent peripheral is the CAN transceiver. The transceiver communicates over the CAN bus to the vehicle via its OBD-II diagnostic port, a port required on US (United States) automobiles for over 25 years at time of writing. The diagnostic port is extremely powerful, allowing for hundreds of signals to be read (and written, in some cases) from the vehicle. In our small-scale demonstration, only throttle position, speed, and computed acceleration are of interest due to time and legality limitations; we cannot accurately demonstrate a poor driver score on accessible roads (although the availability of a closed course will be discussed later). Our proposed hardware implementation goes slightly beyond our Python-oriented demonstration, introducing an IMU for more accurate acceleration values and OLED display for in car use.

Please find a diagram of the HW architecture in the figure below:



The embedded software in the microcontroller has multiple components. Firstly, data must be captured and filtered from the data sources – in this case, the CAN bus and IMU. The CAN bus directly provides continuous streams of the vehicle's throttle position and speed. We compute acceleration in the demonstration, but the IMU can supplement this data to include both longitudinal and latitudinal acceleration. Note we are not concerned with acceleration in the z-domain (parallel to the force of gravity).

Fuzzy logic initializes and modifies the driver score based on these data inputs. Data processing is done to identify characteristics of unsafe driving, such as high throttle position or excessive acceleration. In a purely hardware implementation in the field, the IMU-sourced latitudinal acceleration contributes as well – such that significant steering angle, for example, particularly at speed, also constitutes unsafe driving.

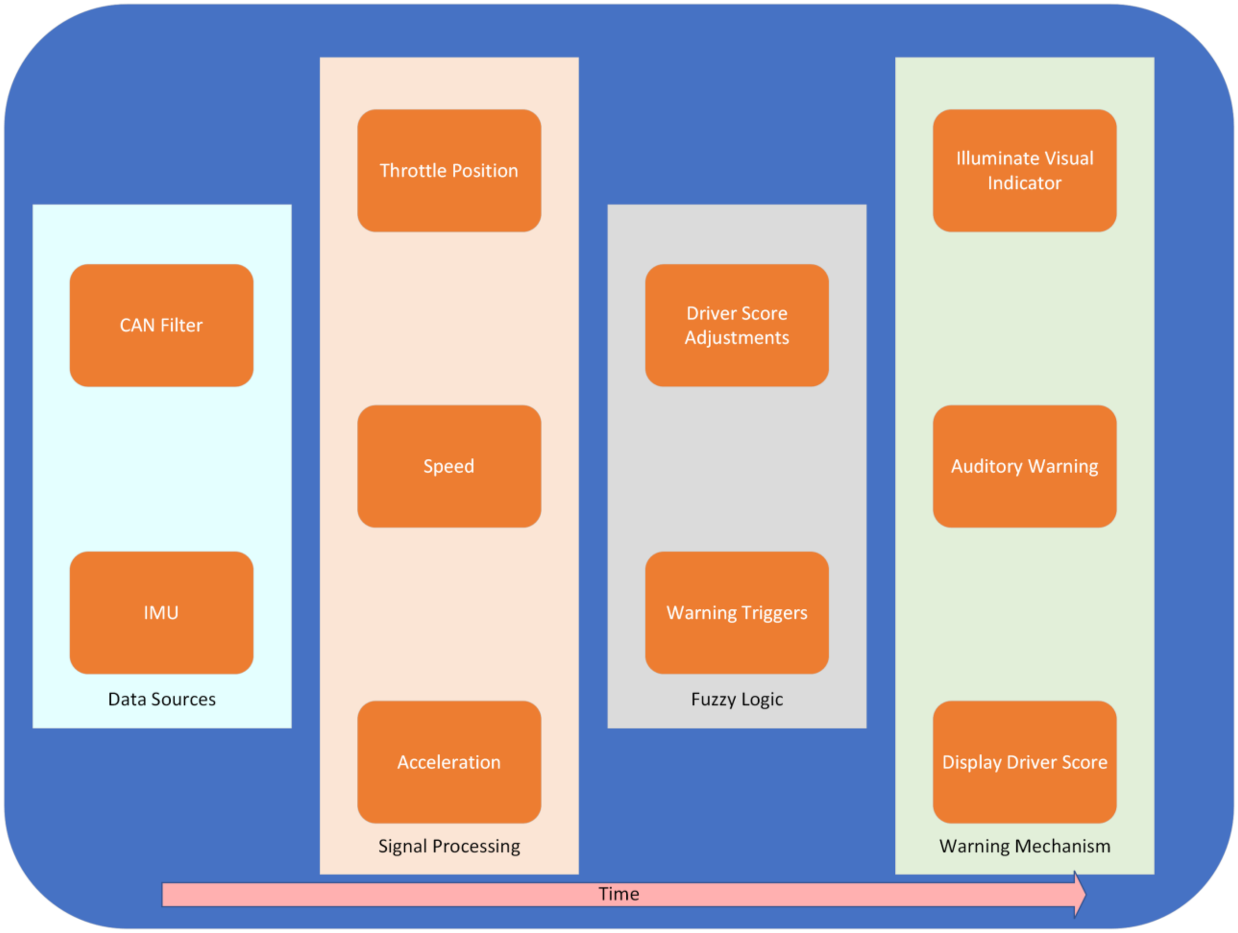
At vehicle startup, the driver score is initialized to 0.5. It ranges from 0 to 1 – while technically discrete, it is modelled to be continuous to a reasonable degree. The driver score continuously updates based on the driver's inputs/behavior, according to the data sources processed before. Low levels of acceleration and throttle position, for example, will raise the driver score accordingly (up to an asymptote). The inverse is also true. As the rolling-average of the driver score changes, warnings are given with varying severity/intensity.

A general scale is presented as follows, implemented via a simple case statement according to the driver score:

- Poor Driver: $0 \leq$ driver score ≤ 0.19; extremely verbose (rapid auditory tone beep/verbal warnings/visual)
- Fair Driver: $0.20 \leq$ driver score ≤ 0.39; verbose warnings (moderate rate auditory tone beep/reduced verbal warnings/visual)
- Average Driver: $0.40 \leq$ driver score ≤ 0.54; stock warnings (slow auditory tone beep/no verbal warnings/visual)
- Good Driver: $0.55 \leq$ driver score ≤ 0.84; (low if any auditory tone/no verbal warning/visual)
- Excellent Driver: $0.85 \leq$ driver score ≤ 1.0; (visual only)

As seen above, the warnings are a combination of auditory, visual, and (in a fully HW implementation) verbal warnings. Auditory warnings can simply be a single tone, or a repeating warning like an unbuckled seatbelt warning. Visual warnings can be modelled by an individual illuminated LED or can be expanded upon further to piggyback onto existing dashboard warning lights (more discussion in the Further Development and Research below). Lastly, a live readout of the updated driver score is displayed via a console readout or on an OLED display – however, concerns about this potentially becoming a distraction (ironic in our study) may push us to remove this feature beyond a purely developmental stage.

Please find a high-level software architectural diagram below:



To implement this architecture in a software simulation, we used python to simulate our hardware as well as several input streams of data from the automobile. As in the case of a real vehicle, the motion of the system is based on the throttle position. Through this simulation we were able to show that at a minimum, it is possible to follow a driver's activities and tailor the vehicle feedback to the driver based on how the driver performs.

Conclusions

Further Development and Research

There are a multitude of avenues to expand based on our preliminary concept and demonstration. Our IMU was too noisy to get accurate, consistent readings of acceleration. When physically mounted properly and using a higher-quality unit, we expect this issue to be resolved. The IMU will allow the inclusion of latitudinal acceleration. We note that latitudinal acceleration greater than 0.5 G may be considered excessive and uncomfortable for passengers, while values above 0.8 G are reserved for extremely aggressive road/canyon driving and/or autocross and track conditions.

Additionally, a higher-level of integration and customization is desired for future work. Using existing dashboard lights for visual indicators may not be entirely appropriate for most users (and their warranties), but it is a concept we wish to explore. We concede that throttle position is an imperfect indicator of acceleration, as engine/motor power and torque can heavily affect the amount of throttle necessary for a given amount of acceleration (a twin-turbo V8 is obviously more powerful than an N/A inline-4). Customization of engine power from a user’s end may be a useful future feature.

Integration of additional vehicular sensors, like forward-facing radar or blind-spot monitoring, presents an additional interesting area of expansion. Radar can identify unsafe following distance and contribute to driver score, while blind-spot detection can allow for warnings to be more targeted – yet these technologies are not ubiquitous and system implementation vary across manufacturers.

Lastly, the use of a closed road and tandem driving would allow a full evaluation of a complete hardware setup. While that is beyond the scope of this project, it will aid further development and fine-tuning.

Bibliography

[1] A. Gross, "Long-Term Use of Advanced Driver Assistance Technologies Can Result in Disengaged Drivers," AAA, 17 12 2019. [Online]. Available: https://newsroom.aaa.com/2019/12/long-term-use-of-advanced-driver-assistance-technologies-can-result-in-disengaged-drivers/. [Accessed 28 April 2022].