ADAS-RL: Adaptive Vector Scaling Reinforcement Learning For Human-in-the-Loop Lane Departure Warning

Armand Ahadi-Sarkani
aahadisa@uci.edu
University of California, Irvine

Salma Elmalaki
salma.elmalaki@uci.edu
University of California, Irvine

Abstract

Multiple sensory modalities are fast becoming a key instrument in the future of the automotive industry. Collision avoidance, lane departure warning, and self-parking are examples of Advanced Driver Assistance Systems (ADAS) that are becoming possible with the adoption of more sensors. Moreover, thanks to the recent advances in mobile computing and wearable devices, the driver is now equipped with advanced sensory systems. This rich sensory environment paves the way to integrate the human factor into the loop of computation of ADAS to provide a personalized experience. In this paper, we introduce **ADAS-RL**, a Reinforcement Learning based algorithm that integrates the behavior and reactions of the driver with the vehicle context to continuously adapt and tune the warning interventions of Lane Departure Warning System (LDW). We validated ADAS-RL against human drivers using CARLA simulator. Our evaluation shows a significant enhancement in the driver experience compared to the standard LDW systems. ADAS-RL shows the ability to track the changes in driving behavior and adapt the frequency of warnings allowing drivers to stay within a reasonable distance (around 1.75m) from lane markings with a significant decrease in the false warnings.

CCS Concepts

• **Computing methodologies** → **Reinforcement learning**; • **Human-centered computing**; • **Computer systems organization** → *Embedded and cyber-physical systems*;

Keywords

Reinforcement Learning, Q-learning, Lane Departure Warning, Advanced Driver Assistance Systems (ADAS), Human-in-the-Loop

ACM Reference Format:

Armand Ahadi-Sarkani and Salma Elmalaki. 2021. ADAS-RL: Adaptive Vector Scaling Reinforcement Learning For Human-in-the-Loop Lane Departure Warning. In *The First International Workshop on Cyber-Physical-Human System Design and Implementation (CPHS21)*, May 18, 2021, Nashville, TN, USA. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3458648.3460008>

1 Introduction

Recent advances in human behavior modeling [3], human intent estimation [13], and human adaptation algorithms [5, 6] open a multitude of opportunities to disrupt the status quo of designing

modern CPS applications. As human-centered applications are permeating every corner of our lives today, it is a pressing need for smart CPS applications to adapt to the physical environment and adapt to human preferences and behavior [4]. Equivalently, the last two decades have seen a growing trend towards efficient ADAS systems fueled by this dramatic increase in sensing capabilities. Despite the ability of the ADAS to collect and analyze a significant amount of sensory data, traditional ADAS typically depends on fixed policies and rule-based models based on the average human response time and driving behavior. However, fixed policies that do not account for variations in human mood, reactions, and expectations, fail to achieve the promised user experience. This opens the question of how to use the monitored human state to design human-in-the-loop ADAS systems that provide a personalized experience. Nevertheless, a central challenge in any human-in-the-loop system is that the human's reaction and behavior are inherently variable which makes designing such systems a daunting process [3].

In this paper, we propose a Reinforcement Learning (RL) based algorithm to enhance the Lane Departure Warning (LDW) system by integrating the human state in the decision-making process. The current LDW system uses the lane markings to detect if the vehicle touched them to issue the warning. However, we argue that the LDW should be issued ahead of time to account for the human response time that varies based on the human state, such as attention level and the driving behavior. Since the human driving behavior is unknown a priori, varies between different drivers, and varies across time for the same driver, the proposed RL algorithm needs to consider all these variations that are not handled by the standard RL algorithms. Accordingly, we propose **Adaptive Vector Scaling Reinforcement Learning** or **ADAS-RL** in short that optimizes the timing and frequency of warnings issued to different drivers.

1.1 Related Work

Recently, there has been renewed interest in predicting the human driving behavior through building prediction models either to replicate the human behavior or for providing a better user experience [8]. In particular, predicting the intention of making the lane departure to reduce the frequency of the false warnings has been proposed by building a prediction model of the driving trajectory based on pre-recorded human driving samples [17]. Modeling the human steering behavior for the driving wheel to predict the driver's desired path was used to replicate the human's maneuvers behavior at different vehicle speeds [14]. Multisample reinforcement learning has been used to predict and adapt to the changes of the human response time to Forward Collision Warnings (FCW) [5]. Separately from human behavior modeling, adaptive lane departure has been studied in the context of cooperative vehicle infrastructure using trajectory tracking [18] and predicting unforeseen scenarios for lane change maneuvers using deep Q-learning [16].

This work is licensed under a Creative Commons Attribution International 4.0 License.
CPHS21, May 18, 2021, Nashville, TN, USA
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8440-7/21/05.
<https://doi.org/10.1145/3458648.3460008>

Figure 1: Human-vehicle interaction using RL for human-in-the-loop ADAS where the environment is both the human and the vehicle. ADAS-RL agent takes actions based on the fusion of the vehicle and human context. These actions are tuned by a reward received from the environment.

Using multiple sensor modalities to provide a better recommendation have been heavily studied in the automotive industry. In particular, detecting the state of attention of the driver by studying the current context of both the physiological state and the vehicle state to issue a warning to decrease the possibility of a crash, for example, Mercedes-Benz attention assist [11]. Besides drowsiness, other researchers looked into studying the driver behavior to measure the level of fatigue [19], inattentiveness/distraction detection [15], and learning the driving behavior (normal, aggressive) [7].

1.2 Paper Contribution

In this paper, we make the following contribution:

- Designing a modified Q-learning agent named “**Adaptive Vector Scaling RL**” or **ADAS-RL** for short that addresses the changes in the human response time in the context of LDW.
- We use the proposed ADAS-RL to design a human-in-the-loop LDW system by integrating CARLA simulator and wearable devices. This system is used to fuse the human context with the vehicle context in a virtualized simulated environment.
- We implemented a proof-of-concept of the proposed ADAS-RL that demonstrates the feasibility of the algorithm. We evaluated our algorithm on human drivers using the designed simulator.

2 ADAS-RL System Architecture

Our system is divided into three main parts as shown in Figure 1, namely, (i) the human context, (ii) the vehicle and environment context, and (iii) the context fusion with adaptation agent. For the rest of the paper, we will focus on lane departure warning and human attention state to describe these three modules.

2.1 Human Context

Central to creating a human-in-the-loop LDW is to infer the current human state. While the paper’s main contribution is *not* to develop a new human context-inference engine, to validate our proposed algorithm, we focused on one human context which is the attention level. Accordingly, we will build upon recent work in literature to infer the attention level using wearable devices. In particular, recent trends in smart rings have led to a proliferation of studies that enhance the inference accuracy of the human’s state due to their convenience and comfort. Accordingly, we exploit Oura Ring [12], a consumer-grade sleep and activity tracker that provides valuable

information about a particular human’s attentiveness and probability for distraction during driving. Oura separates the data it reports into three categories: sleep, activity, and readiness. Data from all of these categories are weighted according to their relevance in categorizing the human’s attentiveness into attentive, distracted, or drowsy. The weighting of these three categories is as follows: sleep and readiness data are equally weighted ($0.375x$), while activity data is less relevant and has less weight ($0.25x$). This information is used to provide the human state to the ADAS-RL agent as shown in Figure 1.

2.2 Vehicle and Environment Context

This module’s objective is to infer the vehicle state within its surrounding environment. In particular, we focus on multiple sensory information: (i) the vehicle’s location with respect to the location of the left and right lanes, (ii) the vehicle’s velocity, (iii) acceleration, (iv) steering angle, (v) braking intensity, and (vi) the speed limit on the current stretch of road. Most of these sensory data can be directly inferred from the information collected by the car front-facing radars and LiDARs along with the car speedometer. Moreover, raw information from these sensors can be captured by scanning the car’s internal CAN bus through the standard OBD-based CAN scanners.

This module’s output is a 12 discretized value of the relative distance to the lane markings and the center of the lane. All distances below 0.55 m from the adjacent lane are considered in a “dangerous” state (very close or almost crossing the lane marking). All distances above 1.6 m from the adjacent lane are considered in a “safe” state (almost at the center of the lane). Similarly, the velocity of the vehicle is quantified into three values with respect to the speed limit, namely (i) moderate, (ii) close to speed limit, and (iii) over-speeding.

2.3 Context Fusion and Adaptation Agent

The adaptation agent is responsible for (i) using both the driver and the vehicle context to—implicitly—infer the likelihood of approaching a dangerous state based on the driver attention level, the response time, and the driving preference, and (ii) alerting the driver early enough to avoid unintended merge into the neighboring lane. However, the response time and the human preferences change over time and unknown a priori due to the intrinsic variation in human reactions to the issued warnings [3, 5]. Hence, the adaption agent needs to continuously tune the best time to issue the LDW. To address these challenges, the adaptation agent should monitor the driver’s reaction to the issued warning. If the driver reacts to an issued warning by correcting the vehicle’s path to the center of the lane before touching the lane marking (dangerous state), the agent “learns” that the issued warning was at the correct time. If the driver does not react to the issued warning, the agent “learns” that it was a false warning. Through this continuous action/reaction cycle, the agent will be tuned towards the best time to issue the warning for a particular driver. This fits naturally within the “Reinforcement Learning” paradigm. Nevertheless, since the human’s reaction is different as motivated earlier, the RL agent needs to account for this human variations to deliver the promised personalized experience. In the subsequent sections, we illustrate our approach to model the human-in-the-loop RL agent.

Figure 2: Markov Decision Process (MDP) for human driver and the vehicle. The states of the MDP correspond to the state of both the driver (attention level) and the vehicle (relative distance to the lane marking and relative velocity to the speed limit)

3 Human-in-the-Loop Reinforcement Learning

To solve the RL problem, we first need to define the environment - the human and the vehicle with its surroundings - that responds to the applied actions by the RL agent. Since the human response time and reaction vary between different drivers and unknown a priori, we model the change in the human and vehicle state as a Markov Decision Process (MDP) with unknown transition probabilities. Accordingly, to solve the MDP to get the best state-action policy without knowing the transitional probabilities, we use the “Q-learning” algorithm that best fits this problem setup. In the Q-learning algorithm, an agent applies an action to the environment and observes its effect per state. The agent chooses an action that maximizes a notion of reward value. The reward value is a quantification of how *good* the taken action by the agent.

State Space S : The state in the MDP is a tuple $S = (D, V, H)$ of three elements:

- D : Twelve discrete values representing the vehicle’s distance from either of the adjacent lanes encoded from 0 to 11 where 0 represents close proximity to the center of the lane, and 11 represents close proximity to lane markings.
- V : Three discrete values representing the vehicle’s current velocity relative to the posted speed limit on a particular stretch of road as described in Section 2.2.
- H : Three discrete values classifying the driver’s state into several categories based on attentiveness data received from a wearable device (Oura Ring) as described in Section 2.1.

As shown in Figure 2, the distance D is separated into D_L and D_R , which represent the distance to the left and right lanes, respectively. The blue arrows represent major state transitions. They are shown once in the figure for clarity. The top left corner state represents the safest state where the vehicle is centered in between the left and right lanes ($D_L = 1.85$ and $D_R = 1.85$), the driver is alert and attentive, and is driving at a moderate/responsible speed. Similarly, the bottom right corner state represents the most unsafe state. Each state transition is associated with a positive (green) or negative (red) reward, independent of the action taken by the agent.

Action Space A : There are two possible actions at a given state S ; to issue a warning or not issue a warning. The agent chooses an action a by using ϵ -greedy to balance exploration and exploitation.

Reward Function R : The reward measures whether the transition between s (the current state) and s' (the next state) was safe. If action a resulted in a transition from an unsafe state to a safe state, the reward is generally positive. Similarly, if action a resulted in a transition from an unsafe state to a safe state, the reward is generally negative. In particular, we use a weighted sum of the difference between the values of D , V , and H , in state s and s' . Accordingly, the reward function R is as follows, where x , y , and z are the weights:

$$R(s, a, s') = x * (d - d') + y * (v - v') + z * h'$$

3.1 Adaptive Vector-Scaling

As motivated earlier, the response time varies from one human to another. Hence, the reward value associated with the state-action transition has to take into account this delay in the received reward. Accordingly, the effect of the taken action does not appear on the next time sample due to the human response time. Moreover, due to the human variation, this delay cannot be fixed because even for the same human, the response time may change over time. Hence, the next state (s') in our reward function is not necessarily at the next time sample. To address this challenge, we propose **adaptive vector scaling** technique to adapt to this change in the human response. Assume that the initial state is S_0 , at a predefined sampling rate T_s , the new states are continually observed from the environment. These states are saved in a dynamic vector initialized at size V . If the algorithm encounters a state that is classified as “safe” while the previous state is classified as “unsafe”, this indicates that the driver has made some major corrective action, such as changing the wheel steering direction towards the center of the lane or became more attentive. In this case, adding states to this vector will stop, and the new vector size is adjusted based on how quickly this corrective action occurred. If this state transition occurred quicker than average for the given driver, the vector increases in size and allows for warnings to be issued less frequently. Conversely, if this state transition occurred slower than average for the given driver, the vector decreases in size and allows for more granularity, with warnings being issued more frequently. The last sampled state in this vector will be the next state s' in the reward function.

Since the response time of a professional and attentive driver is around 0.2s using average speed [1] and around 4s for sleep-deprived driver [9], we choose the sampling time T_s to be 0.1s to ensure that the ADAS-RL agent will capture all the changes in the environment. We initialize V to have a minimum size of 4 states (0.2s) and a maximum of 40 states (4s).

3.2 Algorithm Design

Based on the described MDP model, we design a Q-learning based agent. In Q-learning, an agent applies an action on the environment and observes its effect per state. The agent chooses an action which maximizes a notion of reward value. The reward value is a quantification of how *good* the taken action by the agent using the following Q-learning function, where α and γ are hyperparameters, referring to the learning rate and discount factor respectively:

$$Q(s, a) = Q(s, a) + \alpha[R(s, a, s') + \gamma \max_a Q(s', A) - Q(s, a)]$$

Algorithm 1 ADAS-RL algorithm for Lane Departure Warning Agent.

Hyperparameters: $\alpha, \gamma, \epsilon, x, y, z$
Require:
 $V = \{\text{'moderate'}, \text{'close to speed limit'}, \text{'overspeeding'}\}$
 $H = \{\text{'attentive'}, \text{'distracted'}, \text{'drowsy'}\}$
 $D = \{0, 1, \dots, 11\}$
States $S = (d_1, v_1, h_1), (d_2, v_2, h_2), \dots, (d_n, v_n, h_n)$
Initialize Q table $Q(S, A)$
Actions $A = \{0: \text{'no warning'}, 1: \text{'warning'}\}$
Minimum vector size $V_{min} = 4$
Adaptive vector size $V = 20$ ▷ Initial size
Maximum vector size $V_{max} = 40$
Sampling rate $T_s = 0.1$
procedure LANEDEPARTUREWARNING
 while true **do**
 $S_0 \leftarrow \text{State}()$ ▷ Receive a new state
 $stateVector = []$
 $a \leftarrow \text{chooseAction}(S_0)$ ▷ Using ϵ -Greedy
 $\text{sendActionToClient}(a)$
 for j in range(0, V) **do**
 $S_j \leftarrow \text{State}()$
 $\text{append}(stateVector, S_j)$
 $\text{sleep}(T_s)$
 if $\text{isSafe}(S_j)$ and $\text{isUnsafe}(S_{j-1})$ **then**
 $V \leftarrow V + \text{vectorAdjustment}(V, j)$ ▷ Adapt the vector size
 break
 $R \leftarrow \text{rewardFunction}(S_0, a, S_{V-1})$
 $Q(S_0, a) \leftarrow Q(S_0, a) + \alpha * (R + \gamma * \max_a Q(S_{V-1}, A) - Q(S_0, a))$

As mentioned earlier, to account for the variation in the human response time, the action is not taken every T_s . However, it depends on the size of the adaptive vector V which changes based on the observed state from the environment. The proposed algorithm is summarized in Algorithm 1.

4 Experimental Results

We evaluate ADAS-RL on two steps. The first step is to evaluate that the algorithm can adapt to different human behavior. Hence, we need a controlled experiment that we can repeat to compare between different results. The second step is to evaluate it using real human drivers. To ensure the safety of the human drivers, to reduce liability, and meet the ethical standards of experimenting on human subjects, we decide to use a simulated environment. In this environment, we use CARLA simulator [2] to design multiple road maps and different driving scenarios. We change the LDW system in CARLA with our proposed ADAS-RL.

4.1 Simulated Human Drivers

To model the human behavior in the simulated experiment, we categorize the driver into four categories:

- **Aggressive driver:** upon receiving a LDW, the driver quickly and aggressively steers towards the center of the lane.
- **Calm driver:** upon receiving a LDW, the driver takes more time to react to the warning, and steers less aggressively.
- **Cautious driver:** this driver acts the same as the calm driver, except, before taking a corrective action towards the center of the lane, the driver waits and observes the surroundings.
- **Drowsy driver:** this driver consistently swerves around the lane, gradually decreases the speed, and responds to warnings less effectively than the other drivers.

We changed the steering, the throttle, and the brakes in CARLA to model these driving behavior. Moreover, to add some randomization to the model, all of these drivers first observe their environment and determine whether or not a corrective action is needed, and take this corrective action 95% of the time. Each driver has a different threshold to the lane marking when they consider taking the corrective actions.

4.2 Experiment 1: Tracking Driver Behavior

We run a simulated driving scenario on CARLA for a particular driver for approximately 20 minutes and we log the actions that the ADAS-RL agent took. Figure 3 displays the actions taken by the agent in the last 100 iterations of the algorithm where each iteration took approximately 2s. The y-axis has 12 points representing the 12 distinct values for D as explained in Section 3. In particular, the lower the number, the closer the vehicle is to the center of the lane. Red dots represent that the agent issued a LDW while green dots represent that the agent did not issue a LDW. As observed in Figure 3, the agent learnt to issue the warning at different distances to the lane marking depending on the response of the corresponding driver. In particular, the agent learnt to issue the warning to the aggressive driver starting from $D = 7$, while learnt to issue the warning to the calm driver starting from $D = 5$. Thanks to the adaptive vector scaling technique, the agent learnt that the calm driver needs the warning ahead of time to make a corrective action. Moreover, for the cautious driver, the warnings are generally issued earlier than the aggressive and calm drivers, since the cautious driver takes time to observe the surroundings and thus takes corrective action slower. As for the drowsy driver who makes unsafe maneuver, the agent learnt to issue the warning even at states that may be considered safe for other drivers. This is also can be observed in how the size of the adaptive vector changed over time for the aggressive and calm driver (shown in Figure 4) which is correlated to their respective response time. In particular, if the driver has a long response time to warnings, the vector size increases to adapt for the time it takes the driver to perform their corrective action and vice versa.

To evaluate the learning performance of our agent, we focus on the following metrics:

- **False Negative (FN):** False negatives occur whenever the car violates the lane marking without having a turn signal on as a result of the LDW deciding not to warn the driver early enough. Lower FN reflects better performance.
- **False Positive (FP):** This metric quantifies the rate of false alarms. That is, the rate at which the LDW is triggered unnecessarily. Lower FP reflects better performance.

Indeed, one can argue that FN has severe consequences compared to FP. The results for the four simulated drivers are shown in Figure 5. In particular, for aggressive driver, calm driver, and cautious driver, the FP is decreasing over time. However, for drowsy driver, the decrease in FP is slightly less. This is accounted for the unpredictability of the drowsy behavior.

4.3 Experiment 2: Changes in Driving Pattern

Since human behavior can change over time, we evaluated the ability of the agent to track the changes in the driving pattern. In particular, we conducted an experiment where a driver started by an aggressive behavior and then switched to be a calm behavior,

Figure 3: Time events for the actions taken by the RL agent for different drivers at different distance states D .

Figure 4: Changes in the adaptive vector size over time for aggressive (left) and calm (right) drivers.

and in the end switched back to an aggressive behavior. In Figure 6, we show the frequency of warnings issued over time. The x-axis is time, measured in iterations. Each tick corresponds to 100 iterations, with each iteration taking approximately 2 seconds. Each bar on the x-axis represents a particular encoded distance D (color-coded in the legend). The y-axis represents the number of warnings issued at the given D . This experiment runs for 2800 iterations starting from an unlearned agent. The driver initially starts having a fast response time by being aggressive. As time progressed, the RL agent started to issue warnings more frequently at more unsafe states. Around iteration 1000, the driver slowed down the response time for warnings by becoming calmer (shown with red horizontal bar), and the RL agent responded with issuing warnings more frequently at intermediate states (yellow). In the end, around iteration 2100, the driver returned to a faster response time by having an aggressive behavior (shown with blue horizontal bar), and the agent began to issue warnings less frequently at intermediate states. This result shows the ability of the RL agent to adapt even to the changes in the driving behavior across time for the same human.

4.4 Experiment 3: Manually Controlled

Environment

After gaining confidence in the proposed ADAS-RL using the simulated human models, we asked three real human drivers to try the LDW system that runs our agent¹. To mimic the driving experience, we built the testbed shown in Figure 7. Our participants used a Logitech G920 steering wheel with pressure sensitive pedals (accelerator, brake, and clutch) to drive a car in CARLA. The participant is also wearing an Oura Ring on their ring finger. Data streaming from CARLA (driving specific data) and from Oura ring (human state data) are collected by a Raspberry Pi that runs our ADAS-RL agent for LDW.

In this experiment, each of the participants is asked to drive the car for a duration of approximately 33 minutes (1000 iterations)

¹At the time of preparing this manuscript, COVID-19 pandemic was hindering against conducting experiments with multiple real human subjects. Social distancing and isolation were enforced in the authors' location.

Figure 5: Right-lane distance (RD), false positive warnings (FP), and false negative warnings (FN) for aggressive, calm, cautious, and drowsy drivers.

Figure 6: Frequencies of warnings issued for driver with a different driving pattern.

for two separate experiments on two different days. The first experiment while the driver is attentive, while the second is when the driver is distracted. To simulate distraction, the human drivers are engaged in a conversation that aims at raising their cognition load [10]. As opposed to the automated driving environment, the driving behaviors are not predictable, and the vehicle is driving on different stretches of road and is susceptible to different real-world driving errors.

Figure 7: Testbed setup including steering wheel, pressure sensitive pedals, Oura Ring, and Raspberry Pi interfacing with CARLA.

Figure 8: Actions taken over time by ADAS-RL for three different human drivers.

As shown in Figure 8, ADAS-RL adapts to the unique behaviors of each of the drivers, including when they are distracted. Driver 1, for example, received more issued warnings in some of the earlier states compared to drivers 2 and 3. In general, when each of the drivers is attentive, ADAS-RL issues warnings with less urgency than when they are distracted. Notably, driver 2 seems to act responsibly after receiving more warnings in earlier states and does not approach the lane markings after 13 minutes. Driver 1 and driver 3 do not seem to be responding well to warnings, and warnings are still issued in early states due to their distraction.

5 Conclusion and Future Work

Existing Lane Departure Warning (LDW) systems rely on algorithmic computations performed on sensory data provided by vehicles, and do not adapt to the distinct driving behaviors and physiological states of different humans. Accordingly, in this paper, we proposed ADAS-RL to integrate human factor into the LDW to personalize the timing and frequency of warnings issued to different drivers based on their response time, driving style and attentiveness. By

considering the human into the loop of computation of Reinforcement Learning, ADAS-RL has converged towards issuing warnings sooner to drivers with slower response times to warnings, and later to drivers with faster response times. Moreover, drivers who have been distracted at the wheel have been faced with more frequent warnings, issued more in advance than less distracted drivers. Thanks to the ability of ADAS-RL to track the variations in the human response time, ADAS-RL showed its effectiveness in adapting to different driving behavior and human state for both simulated and real human drivers. Indeed, in the future we want to explore the effect of ADAS-RL in other ADAS applications.

References

- [1] LC Davis. 2003. Modifications of the optimal velocity traffic model to include delay due to driver reaction time. *Physica A: Statistical Mechanics and its Applications* 319 (2003), 557–567.
- [2] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. CARLA: An Open Urban Driving Simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*. 1–16.
- [3] Salma Elmalaki. 2021. Fair-RL: Fairness-aware Human-in-the-Loop Reinforcement Learning for Harnessing Human Variability in Personalized IoT. In *Proceedings of the 6th ACM/IEEE Conference on Internet of Things Design and Implementation*.
- [4] Salma Elmalaki, Yasser Shoukry, and Mani Srivastava. 2018. Internet of Personalized and Autonomous Things (IoPAT) Smart Homes Case Study. In *Proceedings of the 1st ACM International Workshop on Smart Cities and Fog Computing*. 35–40.
- [5] Salma Elmalaki, Huey Ru Tsai, and Mani Srivastava. 2018. Sentio: Driver-in-the-Loop Forward Collision Warning Using Multisample Reinforcement Learning. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*. 28–40.
- [6] Salma Elmalaki, Lucas Wanner, and Mani Srivastava. 2015. Caredroid: Adaptation framework for android context-aware applications. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. 386–399.
- [7] Derick A Johnson and Mohan M Trivedi. 2011. Driving style recognition using a smartphone as a sensor platform. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 1609–1615.
- [8] Stéphanie Lefèvre, Ashwin Carvalho, Yiqi Gao, H Eric Tseng, and Francesco Borrelli. 2015. Driver models for personalised driving assistance. *Vehicle System Dynamics* 53, 12 (2015), 1705–1720.
- [9] Joy Mara. 2020. Drwosy Driving and Automobile Crashes. https://one.nhtsa.gov/people/injury/drowsy_driving1/human/Drowsy.html. Last accessed February 2021.
- [10] Jason S McCarley, Margaret J Vais, Heather Pringle, Arthur F Kramer, David E Irwin, and David L Strayer. 2004. Conversation disrupts change detection in complex traffic scenes. *Human factors* 46, 3 (2004), 424–436.
- [11] Mercedes Benz. 2021. Mercedes attention assist. <https://www.mbusa.com/mercedes/benz/safety#module-3>. [Online; accessed Feb 2021].
- [12] OURA. 2020. OURA Smart Rings. <https://ouraring.com/meet-oura>. Last accessed February 2020.
- [13] Sid Reddy, Anca Dragan, and Sergey Levine. 2018. Where do you think you're going?: Inferring beliefs about dynamics from behavior. In *Advances in Neural Information Processing Systems*. 1454–1465.
- [14] Scott Schnelle, Junmin Wang, Haijun Su, and Richard Jagacinski. 2016. A driver steering model with personalized desired path generation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 47, 1 (2016), 111–120.
- [15] Ashish Tawari, Sayanan Sivaraman, Mohan Manubhai Trivedi, Trevor Shannon, and Mario Toppelhofer. 2014. Looking-in and looking-out vision for urban intelligent assistance: Estimation of driver attentive state and dynamic surround for safe merging and braking. In *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. IEEE, 115–120.
- [16] Pin Wang, Ching-Yao Chan, and Arnaud de La Fortelle. 2018. A reinforcement learning based approach for automated lane change maneuvers. In *2018 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 1379–1384.
- [17] Wenshuo Wang, Ding Zhao, Wei Han, and Junqiang Xi. 2018. A learning-based approach for lane departure warning systems with a personalized driver model. *IEEE Transactions on Vehicular Technology* 67, 10 (2018), 9145–9157.
- [18] Feng You, Ronghui Zhang, Guo Lie, Haiwei Wang, Huiyin Wen, and Jianmin Xu. 2015. Trajectory planning and tracking control for autonomous lane change maneuver based on the cooperative vehicle infrastructure system. *Expert Systems with Applications* 42, 14 (2015), 5932–5946.
- [19] Zhiwei Zhu and Qiang Ji. 2004. Real time and non-intrusive driver fatigue monitoring. In *The 7th International IEEE Conference on Intelligent Transportation Systems Proceedings*. IEEE, 657–662.