DriCon: On-device Just-in-Time Context Characterization for Unexpected Driving Events

Debasree Das, Sandip Chakraborty, Bivas Mitra

Department of Computer Science and Engineering, Indian Institute of Technology Kharagpur, INDIA 721302

Email: {debasreedas1994, sandipchkraborty, bivasmitra}@gmail.com

Abstract—Driving is a complex task carried out under the influence of diverse spatial objects and their temporal interactions. Therefore, a sudden fluctuation in driving behavior can be due to either a lack of driving skill or the effect of various on-road spatial factors such as pedestrian movements, peer vehicles' actions, etc. Therefore, understanding the context behind a degraded driving behavior just-in-time is necessary to ensure on-road safety. In this paper, we develop a system called *DriCon* that exploits the information acquired from a dashboard-mounted edge-device to understand the context in terms of micro-events from a diverse set of on-road spatial factors and in-vehicle driving maneuvers taken. *DriCon* uses the live in-house testbed and the largest publicly available driving dataset to generate human interpretable explanations against the unexpected driving events. Also, it provides a better insight with an improved similarity of 80% over 50 hours of driving data than the existing driving behavior characterization techniques.

Index Terms—Driving behavior, spatial events, context analysis

Fig. 1: *DriCon*: Hardware components and a running instance when a vehicle faced severe jerks

I. INTRODUCTION

With an increase in the traffic population, we witnessed a phenomenal rise in road accidents in the past few years. According to the World Health Organization (WHO) [1], the loss is not only limited to humans but affects the GDP of the country as well. The officially reported road crashes are inspected mostly based on the *macro* circumstances, such as the vehicle's speed, the road's situation, etc. Close inspection of those *macro* circumstances reveals a series of *micro-events*, which are responsible for such fatalities. For example, suppose a driver hit the road divider and faced an injury while driving on a non-congested road. From the macro perspective, we might presume it is due to the driver's amateurish driving skill or the vehicle's high speed. But, it is also possible that some unexpected obstacles (say, crossing pedestrians/animals) arrived at that moment out of sight. The driver deviated from his lane while decelerating to avoid colliding with them. Therefore, recording these *micro-events* are crucial in identifying the reasoning behind such accidents. Such contextual information, or *micro-events*, thus, can help various stakeholders like car insurance or app-cab companies to analyze the on-road driving behavior of their drivers. Interestingly, an app-cab company can penalize or incentivize their drivers based on how they handle such context and take counter-measures to avoid accidents.

A naive solution to extract the context information is to analyze the traffic videos. Notably, CCTV cameras [2] capture only static snapshots of the events concerning the moving vehicles. Existing works [2], [3] use dash-cam videos along with IMU sensor data for manual or partly automated investigation of the accident. Note that, human intervention is error-prone and labor-intensive with higher costs. The situation gets further complicated when multiple events are responsible for the accident. For instance, suppose the preceding vehicle suddenly brakes to avoid collision with a pedestrian or at a run-yellow traffic signal. Consequently, the ego vehicle has to decelerate abruptly, resulting in a two-step chain of responsible events for the unexpected stop. Thus, identifying spatiotemporal interactions among traffic objects are crucial in characterizing the root cause behind such incidents.

Importantly, understanding the contexts behind the degraded driving behavior on the fly is not trivial and poses multiple challenges. **First**, this involves continuous monitoring of the driving behavior of the driver as well as an exhaustive knowledge of various on-road spatial *micro-events*. Expensive vehicles use LiDAR, Radar etc., to sense the driver and the environment [4], [5]; however, app-cab companies are resistant to invest in such high-end vehicles due to low-profit margin. **Second**, depending on the driving maneuvers taken, *temporally interlinking the micro-events* based on the vehicle's interaction with on-road spatial objects is a significant research challenge. For example, if adverse snowy weather is observed on one day, its effect on traffic movements may last till the next day. In contrast, reckless driving would impact only a few other vehicles around and will not be temporally significant after a few minutes. Such temporal impacts of an event would vary depending on the type and space of the event. **Third**, *spatial positions* of the surrounding objects impact the driving maneuver. Precisely, along with temporal dependency, the distance between the ego vehicle and the surrounding

objects plays a vital role. For example, a far-sighted pedestrian might cross the road at high speed, keeping a safe distance, but it is fatal if the distance to the vehicle is low. Existing literature [6], [7] have attempted to identify risky driving, e.g., vehicle-pedestrian interaction, through IMU and video analysis; however, they fail to capture such temporal scaling or the spatial dependency among surrounding objects. **Fourth**, identifying the context in real-time over an edge-device (such as a dashcam) is essential for providing a just-in-time feed-back. But, deploying such a system for context characterization and analysis from multi-modal data over resource-constrained edge-device is not straightforward.

To address these challenges, we propose *DriCon* that develops a smart dash-cam mounted on the vehicle's dashboard to characterize the *micro-events* to provide just-in-time contextual feedback to the driver and other stakeholders (like the cab companies). It senses the maneuvers taken by the ego vehicle through IMU and GPS sensors. In addition, a front camera mounted on the device itself, is used to analyze the relationship between various on-road *micro-events* and the driving maneuvers taken. This facilitates the system to run in each vehicle in a silo and makes it low-cost and lightweight. Fig. 1 shows a snapshot of the hardware components of our system mounted on a vehicle, and an example scenario where *DriCon* generates a live contextual explanation behind a sudden jerk observed in the vehicle. In summary, our contributions to this paper are as follows.

(1) Pilot Study to Motivate Micro-Event Characterization: We perform a set of pilot studies over the Berkeley Deep Drive (BDD) dataset [8], the largest public driving dataset available on the Internet (as of January 16, 2023), to investigate the variations in driving behavior depending on various road types, time of the day, day of the week, etc., and highlight the spatiotemporal *micro-events* causing abrupt changes in driving maneuvers.

(2) Designing a Human Explainable Lightweight Causal Model: The development of *DriCon* relies on the (i) IMU & GPS data to infer the driving maneuvers, and (ii) object detection model & perspective transformation [9] to detect the surrounding objects and their actions to capture various spatial *micro-events*. Subsequently, we identify the spatiotemporal contexts whenever the driving behavior deteriorates during a trip. Finally, we implement Self Organizing Maps (SOMs), a lightweight but effective causal model to capture the spatiotemporal dependency among features to learn the context and generate human-interpretable explanations.

(3) Deployment on the Edge: We deploy the whole architecture of *DriCon* on a Raspberry Pi 3 model, embedded with a front camera, IMU and GPS sensors (Fig. 1). For this purpose, we make both the IMU and visual processing of the data lightweight and delay-intolerant. Following this, the pre-trained model generates recommendations based on the ongoing driving trip and makes it efficient to run live for

just-in-time causal inferences.

(4) Evaluating *DriCon* on a Live System Deployment and with BDD Dataset: We evaluate *DriCon* on our live in-house deployment, as well as on the BDD dataset [8] (over the annotated data [10]), comprising 33 hours and 17 hours of driving, respectively. We obtain on average 70% and 80% similarity between the derived and the ground-truth causal features, respectively, with top-3 and top-5 features returned by the model, in correctly identifying the *micro-events* causing a change in the driving behavior. Notably, in most cases, we observe a good causal relationship (in terms of average treatment effect) between the derived features and the observed driving behavior. In addition, we perform different studies of the resource consumption benchmarks on the edge-device to get better insights into the proposed model.

II. RELATED WORK

Several works have been proposed in the literature on understanding road traffic and its implications for road fatalities. Early research focused on traffic surveillance-based techniques to prevent road accidents. For instance, National Highway Traffic Safety Administration (NHTSA) [3] had recorded statistics about fatal accident cases; TUAT [2] has been collecting video records from taxis and drivers' facial images since 2005 to derive injury instances into several classes along with driving behavior estimation. In India, the source of information behind the causes of traffic injuries is the local traffic police [11]. In contrast, works like [12], [13] learn the crime type and aviator mobility pattern just-in-time from street view images and raw trajectory streams, respectively. Apart from harnessing videos and crowd-sourced information, several works [14], [15] are done on abnormal driving behavior detection by exploiting IMU and GPS data. To prevent fatal accidents, authors [16]–[18] try to alert the drivers whenever risky driving signature is observed, such as lane departure or sudden slow-down indicating congestion. However, they have not looked into the effect of neighboring vehicles or other surrounding factors on various driving maneuvers.

Interaction among the ego vehicle and other obstacles, such as pedestrians, adverse weather in complex city traffic, often affects the vehicle's motion, consequently affecting the driving behavior. Existing studies [19] reveal that road category, unsignalized crosswalks, and vehicle speed often lead to a disagreement among pedestrians to cross the road, leading to road fatalities. A more detailed study [20], [21] focuses on causality analysis for autonomous driving, faces infeasibility in real-time deployment. Moreover, they only use a limited set of driving maneuvers, e.g., speed change only. Particularly, causal inferencing is challenging due to high variance in driving data and spurious correlation [22] between traffic objects and maneuvers. The existing works limit their study by considering only static road attributes or relying on single or multi-modalities from a connected road network system. Such methodologies will not be applicable for a single vehicle in real-time deployment unless connected to the

system. In contrast, leveraging multi-modalities from onboard vehicle sensors can efficiently characterize the continuous and dynamic contexts behind unexpected driving behavior fluctuations. *DriCon* develops a system in this direction.

III. MOTIVATION

In an ideal scenario, two vehicles are likely to follow similar maneuvers under the same driving environment; but this is not the case in reality. Driving behavior varies according to the driver's unique skill set and is influenced by the impact of various on-road events, such as the movement of other heavy and light vehicles, movement of pedestrians, road congestion, maneuvers taken by the preceding vehicle, etc., which we call *spatial micro-events* or *micro-events*, in short. In this section, we perform a set of pilot studies to answer the following questions. (a) **Does a driver's driving behavior exhibit spatiotemporal variations?** (b) **Do all *micro-events* occurrences during a trip similarly impact the driving behavior?** (c) **Does a sequence of inter-dependent *micro-events* collectively influence the driving behavior?** Following this, we analyze the publicly-available open-source driving dataset named Berkeley Deep Drive dataset (BDD) [8] to answer these questions stating the impact of different *micro-events* on the driving behavior. The dataset contains 100k trips crowd-sourced by 10k voluntary drivers over 18 cities across two nations – the USA and Israel. The dataset has been annotated with a driving score on the Likert scale of 1 (worst driving) to 5 (best driving) for each 5-second of driving trips.

A. Variation in Driving Behavior over Space and Time

We first check whether the on-road driving behavior exhibits a spatiotemporal variation. For this purpose, we vary two parameters – road type as the *spatial* parameter (say, "*Highway*", "*City Street*", "*Residential*"), and time of the day as the *temporal* one (say, "*Daytime*", "*Nighttime*", "*Dawn/Dusk*") in the BDD dataset. In this pilot study, we form 9 groups with 30 trips each, in a total of 270, where the trips under a group are randomly picked from the BDD dataset. We plot the distribution of the driving scores over all the trips for each group. From Fig. 2(a), it is evident that the score distribution varies both (a) for a single type of road at different times of the day, and (b) for different types of road at any given time of the day (with $p < 0.05$ reflecting its statistical significance). In the following, we investigate the role played by various *micro-events* behind the variations in driving behavior.

B. Role of Spatial Micro-events

Next, we inspect whether various on-road *micro-events*, which are characterized by the movements of other spatial objects such as "*cars*", "*pedestrians*", "*trucks*", "*buses*", "*motorcycles*", "*bicycles*", etc., impact a driver's driving behavior in the same way across different times of the day. We perform this study by handpicking 30 trips along with their annotated driving scores for both day and night time from the BDD dataset. We compute the volume (say, count) of spatial objects extracted using the existing object detection

algorithm [23] from the video captured during the trip and take the average count of each object for a 5-second time window. Thus, for both daytime and nighttime, we get two time-series distributions, (a) the count of each on-road spatial object captured over the trip video during each time window, and (b) the annotated driving scores at those time windows. Next, we compute the Spearman's Correlation Coefficient (SCC) among these two distributions for day time and night time, respectively. From Fig. 2(b), we infer that mostly all the on-road spatial objects adversely affect the driving behavior (depicting a negative correlation). Cars and pedestrians affect the driving score majorly during the daytime. Whereas, at night time, trucks and buses, along with the cars, impact the driving behavior because heavy vehicles such as trucks move primarily during the nighttime. However, the effect of light vehicles such as motorcycles and bicycles is insignificant due to the dedicated lanes for their movements. This observation is further extended to Fig. 2(c), where the same study is done for weekdays vs. weekends. We extracted the day of the week using already provided timestamps in the BDD dataset and clubbed 30 trips from Monday to Friday for weekdays and 30 trips from Saturday to Sunday for the weekend. From Fig. 2(c), we observe that during the early days of the week, cars, pedestrians, and trucks adversely affect the driving behavior, whereas the impact is less during the weekend. Hence, we conclude that different on-road objects exert diverse temporal effects on the driving behavior.

C. Micro-events Contributing to Sudden Driving Maneuver: Abrupt Stop as a Use-case

Finally, we explore whether multiple inter-dependent *micro-events* can be responsible for a particular driving maneuver that might degrade the driving behavior. For this purpose, we choose *abrupt stop* as the maneuver, which we extract from the GPS and the IMU data (the situations when a stop creates a severe jerkiness [24]). We take 30 trips for each scenario, including daytime, nighttime, weekdays, and weekends. For each scenario, we extract the instances when an abrupt stop is taken and record the corresponding *micro-events* observed at those instances. Precisely, we extract the presence/absence of the following *micro-events*: red traffic signal, pedestrian movements, presence of heavy vehicles as truck & bus, light vehicles as motorcycle & bicycle, and the preceding vehicles' braking action (as peer vehicle maneuver), using well-established methodologies [10], [23]. We compute the cumulative count of the presence of each *micro-events* and the number of abrupt stops taken over all the trips for the four scenarios mentioned above. From Fig. 2(d) and (e), we observe that the red traffic signal, the peer vehicle maneuvers, and heavy vehicles mostly cause an abrupt stop during the nighttime and on weekdays. Therefore, we argue that multiple on-road *micro-events*, such as the reckless movement of heavy vehicles at night, force even an excellent driver to slam on the brake and take an unsafe maneuver.

Fig. 2: (a) Variation of Driving Behavior with respect to Road Type and Time of the Day, (b)-(c) Impact of Spatial *Micro-events* on the Driving Score at Different (i) Time of the Day, (ii) Day of the Week, (d)-(e) Contributing Factors Observed behind Abrupt Stop at Different (i) Time of the Day, (ii) Day of the Week

IV. PROBLEM STATEMENT AND SYSTEM OVERVIEW

A. Problem Statement

Consider that $\mathcal{F}_M$ denotes the set of driving maneuvers and $\mathcal{F}_S$ be the set of spatial *micro-events*. $\mathbb{F}^i$ be the set of temporally-represented feature variables corresponding to the driving maneuvers taken and on-road spatial *micro-events* encountered during a trip i. Let $\mathcal{R}_T^i$ be the driving score at time T during the trip i. We are interested in inspecting the events occurred, representing the feature values $\mathbb{F}^i$, when $|\mathcal{R}_T^i - \hat{\mathcal{R}}_{T-1}^i| > \epsilon$ (ϵ is a hyper-parameter, we set $\epsilon = 1$), reflecting the fluctuations in driving behavior. Here, $\hat{\mathcal{R}}_{T-1}^i = \lceil \text{mean}([\mathcal{R}_1^i, \mathcal{R}_{T-1}^i]) \rceil$ represents the mean driving behavior till $T-1$. The output of the system is a characterization of $\{\mathcal{F}_M^i, \mathcal{F}_S^i\}$, as to whether a fluctuation in the driving behavior is due to the driving maneuvers only ($\mathcal{F}_M^i$) or forced by the spatially causal *micro-events* ($\mathcal{F}_S^i$). Finally, we target to generate the explanations based on $\{\mathcal{F}_M^i, \mathcal{F}_S^i\}$ to give feedback to the stakeholders for further analysis of the driving profile.

B. Feature Selection

Leveraging the existing literature [24], we identified a set of feature variables at timestamp T representing various driving maneuvers $\mathcal{F}_M$ of the ego vehicle. These features are – Weaving ($\mathcal{A}_T^W$), Swerving ($\mathcal{A}_T^S$), Side-slipping ($\mathcal{A}_T^L$), Abrupt Stop ($\mathcal{A}_T^Q$), Sharp Turns ($\mathcal{A}_T^U$), and Severe Jerkiness ($\mathcal{A}_T^J$). Similarly, we consider the following feature variables corresponding to the spatial *micro-events* $\mathcal{F}_S$ – Relative Speed ($\mathcal{S}_T$) and Distance ($\mathcal{D}_T$) between the ego and the preceding vehicle, preceding vehicle's Braking Action ($\mathcal{B}_T$), volume of the peer vehicles in front of the ego vehicle indicating Congestion in the road ($\mathcal{C}_T$), Pedestrian ($\mathcal{P}_T$), and it's speed ($\mathcal{Q}_T$), Traffic light ($\mathcal{L}_T$), Heavy vehicles: {Bus & Truck} ($\mathcal{H}_T$), Type of the Road ($\mathcal{G}_T$), and Weather condition ($\mathcal{W}_T$). Note that, we empirically select these features based on the existing literature and observations from the dataset; additional features can also be incorporated in *DriCon* without losing its generality.

We next broadly introduce our system architecture. *DriCon* captures IMU, GPS, and video data from a dashcam (say, an edge-device) and characterizes the context behind the improved/degraded driving behavior on the fly. The system comprises three components: (a) **Data Preprocessing and** **Feature Extraction**, (b) **Detection of Improved/Degraded Driving Behavior**, and (c) **Identification of Possible Context** (see Fig. 3).

Fig. 3: *DriCon* System Flow and Modeling Pipeline

C. Data Preprocessing and Feature Extraction

The collected IMU and GPS sensor data are prone to noise due to the earth's gravitational force, signal attenuation, and atmospheric interference. Hence, we implement a low-pass filter to eliminate such noises from IMU and GPS to compute inertial features for the extraction of the driving maneuvers ($\mathcal{F}_M$). Next, we preprocess the video data before extracting on-road spatial *micro-events* and their actions ($\mathcal{F}_S$). We up/downsample the acquired videos to a resolution of $960 \times 540p$, preserving the signal-to-noise ratio above 20 dB.

1) Driving Maneuvers - $\mathcal{F}_M$: In order to generate the features corresponding to different driving maneuvers ($\mathcal{F}_M$), we extract the instances of Weaving ($\mathcal{A}_T^W$), Swerving ($\mathcal{A}_T^S$), Side-slipping ($\mathcal{A}_T^L$), Abrupt Stop ($\mathcal{A}_T^Q$), Sharp Turns ($\mathcal{A}_T^U$), and Severe Jerkiness ($\mathcal{A}_T^J$) from the IMU data using standard accelerometry analysis [10], [24].

2) Spatial Micro-events - $\mathcal{F}_S$: Next, we implement the state-of-the-art video data-based object detection algorithms and further fine-tune them based on our requirements, as developing vision-based algorithms is beyond the scope of our work. We leverage the YOLO-V3 [23] algorithm trained on the COCO dataset [25] to detect a subset of traffic objects such as **Pedestrians**, **Cars**, **Buses**, **Trucks**, and **Traffic Lights** (depicted as $\mathcal{F}_S$). Next, we estimate the influence of pedestrians' interactions, the presence of heavy vehicles (buses & trucks), traffic light signal transitions (red, yellow & green), and the cars on the driving behavior of the ego vehicle. Next, we discard the detected objects which depict a confidence score

$< 50\%$ and bounding boxes of area $< 10k$, capturing the fact that the far-sighted traffic objects around the ego vehicle exert marginal impact compared to the near-sighted ones. Additionally, the traffic objects in the mid-way of the road, broadly visible from the driver's dashboard, will be of more influence than the left or right lanes, as the ego vehicle will follow them immediately. Thus, we divide each of the frames into 0.2:0.6:0.2 ratio along the horizontal axis, as left:middle:right lanes. Therefore, we keep the **Pedestrians** $\mathcal{P}_T$, **Cars**, **Heavy Vehicles** as {**Buses & Trucks**} $\mathcal{H}_T$, which have bounding box co-ordinates within the middle lane boundary, and **Traffic Light Signal Transitions** $\mathcal{L}_T$ (Red, Yellow & Green) without the lane information as traffic lights are often positioned on the left and right lanes. Since our pilot study demonstrated that the pedestrians and peer vehicles' action significantly impact the driving maneuvers of the ego vehicle, (a) we extract the **Pedestrian Speed** ($\mathcal{Q}_T$), as well as identify the crossing pedestrians in the mid-way, and (b) we compute the preceding vehicle's **Braking Action** ($\mathcal{B}_T$), and **Congestion** ($\mathcal{C}_T$), as well as detect the **Relative Speed** ($\mathcal{S}_T$) **and Distance** ($\mathcal{D}_T$) variation among the ego and the preceding vehicle. We apply perspective transformation and deep learning methods [9], [26] to infer the above. Finally, the above pipeline runs on each frame where the video is re-sampled to 15 frames-per-second.

D. Detection of Driving Behavior Fluctuations

The crux of *DriCon* is to capture the temporal dependency of various driving maneuvers and spatial *micro-events* when a change in the driving behavior is observed during the trip. For a run-time annotation of the driving behavior, we use an existing study [10] that provides a driving behavior score on the Likert scale $[1-5]$ by analyzing driving maneuvers and other surrounding factors. We divide the trip into continuous non-overlapping time windows of size δ and compute the driving score at the end of every window $\mathcal{U}$ (denoted as $\mathcal{R}_{\mathcal{U}}^P$), using the feature values captured during that window [10]. To quantitatively monitor whether there is a change in the driving behavior during a window $\mathcal{U}$, we compare $\mathcal{R}_{\mathcal{U}}^P$ and $\hat{\mathcal{R}}_{\mathcal{U}}^P = \frac{1}{\mathcal{U}-1}\sum_{i=1}^{\mathcal{U}-1}\mathcal{R}_i^P$ (mean driving score during previous $\mathcal{U}-1$ windows). Suppose this difference is significant (greater than some predefined threshold ϵ). In that case, *DriCon* proceeds towards analyzing the temporal dependency among the feature vectors at different time windows to understand the reason behind this difference.

E. Identification of Possible Context

In the final module, we use the feature vectors at different windows to build the model that identifies which features ($\mathcal{F}_{GEN}$) are responsible for the change in driving behavior during the window $\mathcal{U}$. The model reactively seeks explanations behind such fluctuations by analyzing the effect of the *micro-events* that occurred over the past windows $[1,\cdots,(\mathcal{U}-1)]$ and the present window $\mathcal{U}$. Finally, natural language-based human interpretable explanations are generated and fed back to the stakeholders for further analysis.

V. MODEL DEVELOPMENT

To develop the core model for *DriCon*, we leverage the already extracted features $\mathcal{F} \in \{\mathcal{F}_M \bigcup \mathcal{F}_S\}$ (details in §IV-C) to capture the temporal dependency of the past as well as the present events. In addition, *DriCon* derives the explanation behind the detected events through explanatory features $\mathcal{F}_{GEN}$. For this purpose, we need a self-explanatory model that can capture the spatiotemporal dependency among different driving maneuvers and *micro-events* associated with the on-road driving behavior. We choose a *Self Organizing Map* (SOM) [27] for constructing the model that can exploit such spatiotemporal dependencies with minimum data availability. The major limitation of the classical deep learning models (such as CNN or RNN) stems from the fact that, (i) deep networks consume heavy resources (say, memory), as well as suffer from huge data dependency, and (ii) they act as a black box, hence fail to generate human interpretable explanations behind certain predictions [28]. On the other hand, SOM is able to characterize the *micro-events* in runtime using feature variability and unlabelled data.

Fig. 4: Working Principle of SOM

A. Inferring Explanatory Features using SOM

The key idea behind obtaining the explanatory features is first to discover the spatiotemporal feature dependency. In *DriCon*, we derive so using Kohonen's Self Organizing Map (see Fig. 4), as it is an unsupervised ANN-based technique leveraging competitive learning methods. Since *DriCon* runs on an edge-device, we employ a minimal number of model parameters to expedite the processing without compromising the performance. Precisely, we implement the *codebook* with 147 neurons, spread out over a two-dimensional array of size 7×21 (where 7 is a hyperparameter depending on the maximum influence of the past windows during a trip, 21 corresponds to the number of features in the feature space). These neurons are initialized with a random weight (see Fig. 4(a)), where the weight vector has the same length (of 21) as the feature vector. Next, we represent each trip with a 2D grid of size 8×21 (considering 8 consecutive windows in a trip) to capture the influence of the past windows $[1,\cdots,(\mathcal{U}-1)]$ and the present window $\mathcal{U}$. In principle, the inherent *topological ordering* of SOM groups the similar feature space (in windows $[1,\cdots,(\mathcal{U}-1)]$) into a *single group*, when there is no change in the driving behavior. On the contrary, the dissimilar ones

(say, during the window $\mathcal{U}$), when there exists a change in the driving behavior, are mapped into a *different group*, as depicted in Fig. 4(b,c).

For instance, suppose on a trip, the ego vehicle abruptly stops due to the preceding vehicle's braking action following a sudden change in the traffic signal. Hence the feature space in window $[1, \cdots, (\mathcal{U} - 1)]$ exhibits a similar signature (until the abrupt stop occurs), and subsequently gets mapped to a *single neuron*. However, during the abrupt stop, there will be changes in the feature space (say, maneuvers and other spatial events). These changes in the feature space will get it assigned to a *different neuron* and settle the other neurons' weight automatically depending on the changes in the feature space between the windows $[1, \cdots, (\mathcal{U} - 1)]$ and the window $\mathcal{U}$. This procedure allows SOM to harness the temporal dependency among spatial events in an unsupervised mode, without using the driving score explicitly.

1) Model Training: The input trip data is represented in the $2D$ grid format for learning the best-matched neuron, optimizing the Euclidean distance between the feature space and weight vector of the corresponding neuron. To ensure the best-fitting, the best-matched neuron tries to learn the weight vector of the feature space at most. Also, the neurons in the neighborhood try to tune their weights as nearest as possible compared to the best-matched neuron. We train this model for 500 epochs, where each neuron gets mapped with the best matching trip instances and converges to their coordinate position in the *codebook*. We implement the Bubble neighborhood function [29] to update the neighborhood neurons' weights until the neighborhood radius converges to ≈ 0. We ensure that both the distance and neighborhood functions are computationally faster for accurate learning accelerating the convergence. Upon completing the total number of epochs, we obtain the converged codebook called the *Map*, where each trip instance gets assigned to the best matching neuron called the *Best Matching Unit (BMU)*. The weight vector corresponding to the BMU's coordinate reveals the explanatory features $\mathcal{F}_{GEN}$.

2) Model Execution: We leverage the constructed *Map* for the runtime inference. First, we conduct the feature processing of the current ongoing trip (following §IV-C), and in parallel, the extracted feature space is fed as input to the constructed *Map*. Eventually, we obtain the BMU's coordinate and extract its corresponding weight vector and the feature encoding for the given trip instance. From the weight vector, we extract the top-k weights and their corresponding feature names (say, *weather type*) and their encoded values (say, *weather type: rainy*). Finally, we populate them in $\mathcal{F}_{GEN}$ (called the *Generative micro-events*) for further generation of human interpretable explanation.

B. Generating Textual Explanation

DriCon aims to generate the explanations in textual format utilizing the output features $\mathcal{F}_{GEN}$ for better readability and human interpretation. As the features $f \in \mathcal{F}_{GEN}$ are already associated with some keywords (say, *severe jerkiness*), we need to generate them in a sentential form, keeping the features as "action" or "subject" depending on whether $f \in \mathcal{F}_M$ or $f \in \mathcal{F}_S$, respectively. For instance, if the feature is an *action*, we assign the ego vehicle as the subject, replace the corresponding output feature f with its describing keyword, and finally concatenate them to obtain the sentential form. For example, in case of *severe jerkiness*, the constructed sentence becomes, "*the ego vehicle severe jerks*". However, if the output feature f represents a *subject*, then many possible sentences can be generated out of one subject. Thus, we mine several traffic guidelines [30] and compute the cosine similarity among the features and existing guidelines using TF-IDF vectorizer. Upon extracting the most relevant guidelines, we fetch the object associated with the sentence and construct a single sentence for each output feature (e.g., "*pedestrian crossing*" → "*pedestrian crossing the intersection*"). Next, for all the generated sentences, the describing keywords corresponding to each feature are converted to an adjective or adverb using Glove [31] for better structuring of the sentences (say, "*the ego vehicle severe jerks*" → "*the ego vehicle severely jerks*"). Finally, each sentence is concatenated using the "*and*" conjunction, and repetitive subjects are replaced using their pronoun form using string manipulation to generate the whole explanation, as depicted in Fig. 3(e).

VI. Performance Evaluation

This section gives the details of *DriCon* implemented over a live setup as well as over the BDD dataset. We report the performance of the SOM model and compare it against a well-established baseline. Additionally, we show how well our system has generated the textual explanations along with a sensitivity analysis to distinguish how error-prone *DriCon* is. We start with the experimental setup details as follows.

A. Experimental Setup

DriCon is implemented over a Raspberry Pi 3 Model B microprocessor kit operating Raspbian OS with Linux kernel version $5.15.65 - v7+$ along with 1 GB primary memory and ARMv7 processor. We primarily utilize the IMU, the GPS, and the video data captured through the front camera (facing towards the front windscreen) as different modalities. For this purpose, we embed one MPU-9250 IMU sensor, one u-blox NEO$-6M$ GPS module, and one Logitech USB camera over the Raspberry Pi board, as depicted in Fig. 1(a). We deployed *DriCon* over three different types of vehicles (e.g., SUV, Sedan, & Hatchback). We hired 6 different drivers in the age group of $[20 - 45]$ who regularly drive in practice. Therefore, our whole experimentation ran for more than two months over three cities, resulting in approximately 33 hours of driving over 1000 km distance. The drivers drove freely without any specific instructions given, with each trip varying from approximately 20 minutes to 2 hours. In addition, each driver drove over five different types of roads (city street, highway, residential, parking & campus road) at three different times of the day (day, dusk & night). We evaluate *DriCon* by analyzing how well our proposed model extracts the generative

micro-events $\mathcal{F}_{GEN}$ (see §V-B). For implementing *DriCon*, we consider $\delta = 5$ seconds, $\epsilon = 1$. The impact of other hyperparameters and resource consumption have been discussed later during the analysis. We next discuss the ground-truth annotation procedure used for the evaluation of *DriCon*.

B. Annotating Micro-events

We launched an annotation drive by floating a Google form among a set of recruited annotators, where they had to watch a video of at most 10 seconds and choose the top-3 most influential factors impacting the driving behavior. We do this annotation over the in-house data (video data collected during the live experiments) and the videos over the BDD dataset. For each video from both the datasets given in the form, we showed only the clipped portion where the score fluctuations had occurred. Next, out of the total 15 factors (including driving maneuvers and spatial *micro-events*) given in a list, they were instructed to choose the top-3 most influential factors responsible for the poor driving behavior based on their visual perception. Besides, we also provided the model-generated sentences (§V-B) and asked how relevant and well-structured the sentences are (on a scale of $[1-5]$) for explaining the change in the driving behavior. The annotators also had the option to write their own explanation if they perceived a better reason behind the driving behavior change. As the number of trips is quite large, we need to design a set of Google forms (sample form[1]), each containing at most 20 videos to ensure the least cognitive load on the annotators. We also collected annotators' demographic information such as age, gender, city, etc. We find that most participants ($> 67\%$) had prior driving skills. At least three independent annotators had annotated each instance. Upon receiving the annotated factors, we need to find the agreement among the annotators to ensure the received ground truth is unbiased and non-random. As standard inter-annotator agreement policies (say, Cohen's kappa index) work on quantitative analysis or one-to-one mapping, we cannot apply such metrics. Thus, we use the majority voting technique where each listed factor is assigned a percentage, signifying how many times the annotators choose that factor. Each factor having a vote of at least 60% is kept in $\mathcal{F}_{GT}$. We observe the minimum and the maximum cardinality of $\mathcal{F}_{GT}$ are 3 and 5, respectively. This also indicates that the annotators agreed on selecting the factors that influenced the driving behavior. $\mathcal{F}_{GT}$ contains the annotated *micro-events* against which $\mathcal{F}_{GEN}$ is evaluated.

C. Performance Metric

We use the **Dice Similarity Coefficient score** [32] ($\mathcal{N}$) which computes the similarity between $\mathcal{F}_{GT}$ and $\mathcal{F}_{GEN}$ as follows: $\mathcal{N} = \frac{2 \times |\mathcal{F}_{GT} \cap \mathcal{F}_{GEN}|}{|\mathcal{F}_{GT}| + |\mathcal{F}_{GEN}|}$. We report the mean $\mathcal{N}$ across all the trips to measure the accuracy of *DriCon*. Next, we also use **Average Treatment Effect** [33] (ATE) to report comparatively higher causal features out of the model identified features. Finally, we define **Percentage of Error** as follows. First, we

Fig. 5: (a) Dice Coefficient Similarity (in %) between Human Annotated and Model Generated Features (b) Ablation Study

compute the set-difference as $\{\mathcal{F}_{GT} \setminus \mathcal{F}_{GEN}\}$, and extract the corresponding feature category (say, $\mathcal{F}_M$, $\mathcal{F}_S$). Once we get the count of each feature category, we compute its percentage out of the total trips as the **Percentage of Error**.

D. Baseline Implementation

As a baseline for extracting $\mathcal{F}_{GEN}$, we implement a supervised rule-based Random Forest (RF) algorithm with 20 decision trees where each tree is expanded to an unlimited depth over the training data. We optimize the labels $\mathcal{R}_{\mathcal{U}}^P$ with the intuition that features will contribute differently to each of the predicted scores. Although the RF-based model has a feature importance score signifying the contribution of each feature in constructing the model, we need to have an explanation of how each feature contributes to predicting the driving scores on a trip instance basis. Therefore, we use LIME [34] in the background of the RF model for generating the explanatory features. As LIME is a model-agnostic method, it tries to map the relationship between the input features and output scores by tweaking the feature values. Thus, it explains the range of values and probability for each feature that contributes to predicting the score. From the generated explanation, we extract the contributing features $\mathcal{F}_{GEN}$ along with their values for further generation of textual explanation. This pipeline is executed in a similar manner as described in §VI-A.

E. Accuracy of Characterized Context

We present the accuracy of *DriCon* using the SOM and RF+LIME model over the in-house dataset using Dice Coefficient Similarity $\mathcal{N}$. We extract the top-k features from $\mathcal{F}_{GEN}$ where $k \in \{3, 5\}$ and compute $\mathcal{N}$ between the two sets of features – $\mathcal{F}_{GEN}$ and $\mathcal{F}_{GT}$ with top-k. Fig. 5(a) shows the result. For top-3, we get 69% & 40% similarity on average with SOM and RF+LIME, respectively. Whereas for top-5, we observe 79% & 48% similarity on average with SOM and RF+LIME, respectively. As the in-house dataset has more complex *micro-events*, the slight performance drop over the in-house dataset using the top-3 features is tolerable. Intuitively, the model can capture more diversity as perceived by the human annotators; therefore, the similarity improves as we move from $k = 3$ to $k = 5$. However, as the RF+LIME considers each time instance of a trip independently, its performance degrades. It captures the dominant features responsible for the driving behavior change within the current time window, contrary to inspecting past time windows' impact.

TABLE I: Similarity Measure among Human Annotated vs. Model Generated Output

Instance#	Human Annotated $\mathcal{F}_{GT}$	Model Generated $\mathcal{F}_{GEN}$	Similarity $\mathcal{N}(\%)$	ATE
1	Poor Weather Conditions (Heavy Rainfall, Fog, etc.), Swerving, **Congestion**, Overtaking, **Taking Abrupt Stop**	**Congestion**, Preceding Vehicle Braking, Weaving, **Abrupt Stop**, Severe Jerkiness	40%	1.96
2	Sideslip, **Taking Abrupt Stop**, **Traffic Lights: Red**	**Traffic Lights: Red**, Congestion, **Abrupt Stop**	66.67%	2.5
3	**Crossing Pedestrian**, High Speed Variation among Cars, **Weaving**	Severe Jerkiness, **Crossing Pedestrian**, **Weaving**	66.67%	1.35

To have a glimpse, we present the explanatory features ($\mathcal{F}_{GEN}$) vs. human-annotated ones ($\mathcal{F}_{GT}$) in Table I for a sample of three test instances where the similarity (Dice coefficient) is comparatively lower. Interestingly, when there is a mismatch, we observe that the corresponding features from the model-generated and human-annotated ones are conceptually related for most of the time. Additionally, a positive high mean ATE value for the model-generated mismatched features signifies that the model perceived those features as more causal than normal human perception. It can be noted that an ATE value ≥ 1 indicates high causal relationships between the features and the corresponding effect (changes in the driving behavior). For example, in test instance #2, the mismatched features are *Sideslip* (for human generated) and *Congestion* (for model generated), where *Congestion* was relatively more causal, affecting the change in the driving behavior. By manually analyzing this instance and interviewing the corresponding driver, we found that he indeed made a minor sideslip on a congested road. Indeed, the driver was not very comfortable in driving a manually-geared car on a congested road.

Fig. 6: Generated Map from SOM for a 7×7 Network (Scaled Down)

F. Ablation Study

Next, we understand the importance of different feature categories corresponding to the driving maneuvers and on-road spatial events, as described in §IV-A, on the overall performance of *DriCon*. To study the impact of driving maneuvers and spatial features, we implement SOM, excluding each of the above feature classes one at a time, and evaluate $\mathcal{N}$ to inspect the importance of each. The two variants other than *DriCon* are constructed in the following way. **(a) *DriCon-man.:*** Here, we exclude the driving maneuvers $\mathcal{F}_M$ and keep

$\mathcal{F}_S$ only. **(b) *DriCon*-spat.:** Next, we exclude the spatial features $\mathcal{F}_S$ and keep $\mathcal{F}_M$ only. We evaluate these two variants over both top-3 and top-5 generated features, along with *DriCon* containing all the features, as depicted in Fig. 5(b). On excluding the driving maneuvers and spatial features, performance drops to 45% and 31%, respectively, for top-5 features. This drastic drop signifies the crucial importance of spatial features, as these are the frequently changing features responsible for fluctuating driving behavior.

G. Model Insight

To understand how the spatiotemporal dependency among different features corresponding to the driving maneuvers and various on-road spatial *micro-events* are derived, we use 49 neurons spread over a 7×7 two-dimensional array (a smaller variant of the SOM network originally used to develop the model, as the original model having 147 neurons is difficult to visualize), fitted over 200 trips. This instance produces a *Map* as depicted in Fig. 6, where all the given trips are assigned to each of the neurons. The scores $\mathcal{R}_{\mathcal{U}}^{P}$ are used only for visual depiction purpose of how the trips are located on the **Map**. Each trip captures the change in the driving behavior using the feature variation. The neurons with multi-color are of more importance than the mono-color, as in those, the score fluctuations are most observed. During a stand-alone trip, the features corresponding to each instance of the trip will have a similar value until there is a change in the driving behavior, thus getting assigned to the same neuron (mono-color). However, the difference in the driving behavior induces distinct feature values than the previous instances; thus, it gets assigned to a different neuron in the *Map*. The neurons having multi-color, as depicted in Fig. 6, map the trip instances where a sudden change of driving behavior has occurred.

H. Dissecting DriCon

We next benchmark the resource consumption behavior of *DriCon*, followed by an analysis of the model's significance and sensitivity.

*1) **Edge-device Resource Consumption***: We benchmark the CPU & memory usage, processing time, temperature rise, and energy consumption over two cases: when (a) the device is idle, & (b) *DriCon* is running. From Fig. 7(a), we observe that in idle mode, on average, 2% of CPU (using "top" command) is used. In contrary, running *DriCon* acquires at most 10% of the processor, which is acceptable. However, the memory usage is a bit high ($\approx 500MB$) mainly due to video processing overhead as depicted in Fig. 7(b). Next, we show the required processing time starting from data acquisition to output generation on a number of trip basis.

Fig. 7: Resource Consumption over the Edge-device (a) CPU Usage (b) Memory Usage (c) Histogram of Processing Time w.r.t., #Trips (d) Temperature Rise, (e) Energy Consumed

Fig. 8: (a) Significance of *DriCon* (b) Sensitivity Analysis of *DriCon* (c) Performance on BDD Dataset

DriCon generates the output within ≈ 3 minutes only for majority of the trips, further validating shorter response time (see Fig. 7(c)). To further delve deeper, we also log the temperature hike (from "vcgencmd measure_temp" command) and total energy consumption using Monsoon High Voltage Power Monitor [35] while running *DriCon*. From Fig. 7(d) & (e), we observe that the temperature hiked at most to 59°C, while on average, 13 Watt-hour energy is consumed, which is nominal for any live system. To benchmark *DriCon*, we have also measured the energy consumption of the Nexar dashcam, which consumes 22 Watt-hour on an average, while capturing very few driving maneuvers (say, hard brake) without any context. This further justifies that *DriCon* never exhausts the resources on the edge-device and is can accurately detect the *micro-events* precisely.

2) Significance of Generated Explanation: Next, we check how significant our generated explanations are. As reported in §VI-B, we plot the distribution of annotated scores (given by the recruited annotators) for the two fields – "*Relevance*" and "*Well-Structured*". "Relevance" signifies the generated explanation's applicability in explaining unexpected events. In contrast, "Well-structured" indicates how well interpretative the generated sentences are as per human cognition. Fig. 8(a) depicts a median value of 5 and 4 for "Relevance" and "Well-Structured", respectively, which further justifies the credibility of *DriCon*. We also compute the similarity between the human-annotated and model-generated sentences and obtain a minimum, maximum, and mean similarity value as 51.33%, 85.5% & 70.57%, respectively, using the TF-IDF vectorizer. Thus *DriCon* resembles human cognition level up to an indistinguishable level (between a human and model) of auto-generating a contextual explanation, which further shows its applicability to give feedback to the stakeholders for their decision-making procedure.

3) Sensitivity of DriCon: Finally, we inspect the micro-events that *DriCon* fails to capture. Because, apart from a model's efficiency, we must also look into its deficiency to analyze how much that might affect the overall performance. Especially, this is important in the case where stakeholders are boosting/penalizing the driver's profile. As depicted in Fig. 8(b), incompetence to capture both the spatial and maneuvers is low. Although this might lead to degraded model performance, as studied in §VI-F; driving maneuvers ($\mathcal{F}_M$) do not contribute superiorly to model performance due to the inter-dependency on spatial features ($\mathcal{F}_S$). But for $\mathcal{F}_S$, the **Percentage of Error** is still $\leq 13\%$, making the system less sensitive into generating error-prone contextual explanations.

I. Offline Performance

Finally, we report the accuracy of our system over the BDD dataset comprising 17 hours of driving data over $1.5k$ trips using $\mathcal{N}$. As depicted in Fig. 8(c), *DriCon* performs quite well on pre-recorded data, with $\mathcal{N} = \{71\%, 84\%\}$, for top-3 and top-5 features. We observe that SOM can identify the *micro-events* in a better way for offline analysis with a public dataset. However, as running the system live is essential for a realistic driving environment other than offline analysis, this much of slight accuracy drop can be endured.

VII. CONCLUSION

This paper developed an intelligent system on the edge-device called *DriCon* leveraging multi-modalities to detect the *micro-events* responsible for unexpected fluctuations in driving behavior. The human-interpretable explanations generated by *DriCon* show their relevance and credibility in identifying such context. Further, the spatiotemporal dependency among various features is inspected in an unsupervised manner to capture a diverse set of driving scenarios. Additionally, the resource-friendly deployment over a live testbed further validates *DriCon*. Although our study captures the context where each feature's contribution is taken independently, inter-feature dependency is not captured explicitly. For instance, say, a driver suddenly weaves while taking a turn to avoid colliding with a crossing pedestrian, making the following vehicle's driver slam the brake. Here, the first driver's action is due to the crossing pedestrian, which in turn impacts the second driver's action. The analysis of such complex and collective interactions among the vehicles needs a more sophisticated

system, possibly a different modality that can connect the inter-vehicle interactions. However, *DriCon* provides a simple, in-the-silo solution that can be independently deployed over vehicles with a dashboard-mounted edge-device or dashcam.

REFERENCES

[1] "Road traffic injuries, by world health organization (who)," https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries, 2022, (Online Accessed: January 16, 2023).

[2] "Institute of engineering tokyo university of agriculture and technology (tuat). smart mobility research center - research." https://web.tuat.ac.jp/~smrc/research.html, 2017, (Online Accessed: January 16, 2023).

[3] N. H. T. S. A. (NHTSA)., https://www.nhtsa.gov/, (Online Accessed: January 16, 2023).

[4] "Lidars for self-driving vehicles: a technological arms race," https://www.automotiveworld.com/articles/lidars-for-self-driving-vehicles-a-technological-arms-race/, 2020, (Online Accessed: January 16, 2023).

[5] Z. Li, C. Wu, S. Wagner, J. C. Sturm, N. Verma, and K. Jamieson, "Reits: Reflective surface for intelligent transportation systems," in *22nd ACM HotMobile*, 2021, pp. 78–84.

[6] R. Akikawa, A. Uchiyama, A. Hiromori, H. Yamaguchi, T. Higashino, M. Suzuki, Y. Hiehata, and T. Kitahara, "Smartphone-based risky traffic situation detection and classification," in *IEEE PerCom Workshops*, 2020, pp. 1–6.

[7] D. A. Ridel, N. Deo, D. Wolf, and M. Trivedi, "Understanding pedestrian-vehicle interactions with vehicle mounted vision: An lstm model and empirical analysis," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 913–918.

[8] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving video database with scalable annotation tooling," *arXiv preprint arXiv:1805.04687*, vol. 2, no. 5, p. 6, 2018.

[9] "Vehicle detection and distance estimation," https://towardsdatascience.com/vehicle-detection-and-distance-estimation-7acde48256e1, 2017, (Online Accessed: January 16, 2023).

[10] D. Das, S. Pargal, S. Chakraborty, and B. Mitra, "Dribe: on-road mobile telemetry for locality-neutral driving behavior annotation," in *23rd IEEE MDM*, 2022, pp. 159–168.

[11] D. Mohan, G. Tiwari, and K. Bhalla, "Road safety in india: Status report 2019. new delhi: Transportation research & injury prevention programme, indian institute of technology delhi." http://tripp.iitd.ac.in/assets/publication/Road_Safety_in_India2018.pdf, 2019, (Online Accessed: January 16, 2023).

[12] K. Fu, Z. Chen, and C.-T. Lu, "Streetnet: preference learning with convolutional neural network on urban crime perception," in *Proceedings of the 26th ACM SIGSPATIAL*, 2018, pp. 269–278.

[13] K. Patroumpas, N. Pelekis, and Y. Theodoridis, "On-the-fly mobility event detection over aircraft trajectories," in *Proceedings of the 26th ACM SIGSPATIAL*, 2018, pp. 259–268.

[14] I. Janveja, A. Nambi, S. Bannur, S. Gupta, and V. Padmanabhan, "Insight: monitoring the state of the driver in low-light using smartphones," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 3, pp. 1–29, 2020.

[15] X. Fan, F. Wang, D. Song, Y. Lu, and J. Liu, "Gazmon: eye gazing enabled driving behavior monitoring and prediction," *IEEE Transactions on Mobile Computing*, 2019.

[16] M. Walch, M. Woide, K. Mühl, M. Baumann, and M. Weber, "Cooperative overtaking: Overcoming automated vehicles' obstructed sensor range via driver help," in *11th ACM AutomotiveUI*, 2019, pp. 144–155.

[17] H. T. Lam, "A concise summary of spatial anomalies and its application in efficient real-time driving behaviour monitoring," in *Proceedings of the 24th ACM SIGSPATIAL*, 2016, pp. 1–9.

[18] S. Moosavi, B. Omidvar-Tehrani, R. B. Craig, A. Nandi, and R. Ramnath, "Characterizing driving context from driver behavior," in *Proceedings of the 25th ACM SIGSPATIAL*, 2017, pp. 1–4.

[19] Y. Shi, R. Biswas, M. Noori, M. Kilberry, J. Oram, J. Mays, S. Kharude, D. Rao, and X. Chen, "Predicting road accident risk using geospatial data and machine learning (demo paper)," in *Proceedings of the 29th ACM SIGSPATIAL*, 2021, pp. 512–515.

[20] M. R. Samsami, M. Bahari, S. Salehkaleybar, and A. Alahi, "Causal imitative model for autonomous driving," *arXiv preprint arXiv:2112.03908*, 2021.

[21] V. Ramanishka, Y.-T. Chen, T. Misu, and K. Saenko, "Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning," in *IEEE CVPR*, 2018, pp. 7699–7707.

[22] F. Codevilla, E. Santana, A. M. López, and A. Gaidon, "Exploring the limitations of behavior cloning for autonomous driving," in *IEEE/CVF ICCV*, 2019, pp. 9329–9338.

[23] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.

[24] J. Yu, Z. Chen, Y. Zhu, Y. Chen, L. Kong, and M. Li, "Fine-grained abnormal driving behaviors detection and identification with smartphones," *IEEE Transactions on Mobile Computing*, vol. 16, no. 8, pp. 2198–2212, 2016.

[25] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*. Springer, 2014, pp. 740–755.

[26] D. Das, S. Pargal, S. Chakraborty, and B. Mitra, "Why slammed the brakes on? auto-annotating driving behaviors from adaptive causal modeling," in *IEEE PerCom Workshops*, pp. 587–592.

[27] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[28] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[29] "Neighborhood function," https://users.ics.aalto.fi/jhollmen/dippa/node21.html, (Online Accessed: January 16, 2023).

[30] "Guidelines for pedestrian facilities," http://www.irc.nic.in/admnis/admin/showimg.aspx?ID=345, (Online Accessed: January 16, 2023).

[31] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *EMNLP*, 2014, pp. 1532–1543.

[32] A. Carass, S. Roy, A. Gherman, J. C. Reinhold, A. Jesson, T. Arbel, O. Maier, H. Handels, M. Ghafoorian, B. Platel *et al.*, "Evaluating white matter lesion segmentations with refined sørensen-dice analysis," *Scientific reports*, vol. 10, no. 1, pp. 1–19, 2020.

[33] D. B. Rubin, "Estimating causal effects of treatments in randomized and nonrandomized studies." *Journal of educational Psychology*, vol. 66, no. 5, p. 688, 1974.

[34] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD*, 2016, pp. 1135–1144.

[35] "Monsoon high voltage power monitor," https://www.msoon.com/online-store/High-Voltage-Power-Monitor-p90002590, (Online Accessed: January 16, 2023).