Expert Systems With Applications 149 (2020) 113240

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Driver behavior detection and classification using deep convolutional neural networks

Mohammad Shahverdy[a], Mahmood Fathy[a,*], Reza Berangi[a], Mohammad Sabokrou[b]

[a] *Department of Computer Engineering, IRAN University of Science and Technology (IUST), Tehran, IRAN*
[b] *Computer Science School, Institute for Research in Fundamental Science (IPM), Tehran, IRAN*

A R T I C L E I N F O

Article history:
Received 6 August 2019
Revised 9 December 2019
Accepted 23 January 2020
Available online 23 January 2020

Keywords:
Driver behavior
Recurrence plot
Convolutional neural networks
Deep learning

A B S T R A C T

Driver behavior monitoring system as Intelligent Transportation Systems (ITS) have been widely exploited to reduce the traffic accidents risk. Most previous methods for monitoring the driver behavior are rely on computer vision techniques. Such methods suffer from violation of privacy and the possibility of spoofing. This paper presents a novel yet efficient deep learning method for analyzing the driver behavior. We have used the driving signals, including acceleration, gravity, throttle, speed, and Revolutions Per Minute (RPM) to recognize five types of driving styles, including normal, aggressive, distracted, drowsy, and drunk driving. To take the advantages of successful deep neural networks on images, we learn a 2D Convolutional Neural Network (CNN) on images constructed from driving signals based on recurrence plot technique. Experimental results confirm that the proposed method can efficiently detect the driver behavior.

1. Introduction

Nowadays, increasing use of vehicles by people has negative consequences such as traffic, accidents, injuries, fatalities, and financial losses. The U.S. Department of Transportation's Fatality Analysis Reporting System (FARS, 2017) has announced that 34,439 fatal accident with 37,461 deaths occurred in the USA in 2016. The mistakes of drivers, and human factors affected by fatigue, alcohol, reckless or careless are the main factors associated with the most accidents. Information about driver, environment, and vehicle can improve the safety of deriving, especially when an abnormal situation occurred on the road. ITSs and Vehicular Ad-Hoc Networks (VANETs) tries to reduce the accident risk by analyzing such information.

Driver behavior is one of the most important factors, which affects road safety. Hence, recently monitoring and detecting systems of driver behavior have been an active research area. Some systems monitor the driver behavior in isolation (Galarza, Egas, Silva, Velasco, & Galarza, 2018; Jo, Lee, Park, Kim, & Kim, 2014), whereas others monitor status of the driver by combining of the driver behavior, the vehicle state and the environment(Al-Sultan, Al-Bayatti, & Zedan, 2013; Daza et al., 2011). Almost most driver behaviour monitoring systems just detect one abnormal behavior (Carmona, García, Martín, Escalera, & Armingol, 2015; Dai, Teng, Bai, Shen, & Xuan, 2010; González, Wilby, Díaz, & Ávila, 2014; Lee & Chung, 2012), whereas few of them detect more than one (Al-Sultan et al., 2013). However, there is still no effective monitoring system to accurately detect all abnormal behaviors of a driver.

So far, many studies have been done on driver behaviors, which monitor the drivers' body and use the deep learning techniques to classify their activities (Danisman, Bilasco, Djeraba, & Ihaddadene, 2010; Galarza et al., 2018; Jo et al., 2014; Sabet, Zoroofi, Sadeghniiat-Haghighi, & Sabbaghian, 2012). Due to the violation of privacy and the possibility of spoofing in camera-based systems some studies desire to use the non-visual driving signals and traditional learning techniques such as Support Vector Machine (SVM) and k-Nearest Neighbors (KNN) (Al-Sultan et al., 2013; Carmona et al., 2015; Dai et al., 2010; Tango & Botta, 2013; Yu et al., 2017).

Deep learning techniques especially CNNs have been successfully developed for image applications. There are many successful deep networks for image classification such as AlexNet (Russakovsky et al., 2015), ResNet (Zeiler & Fergus, 2013), VGGNet (Simonyan & Zisserman, 2014), whilst deep learning techniques have a slow progress for signal processing. Consequently, to take the advantages of such CNNs (*i.e.*, image based ones) for analysing the driver behaviors, we propose to represent the driving signals as several images. To this end, we use the recurrence plot technique (Spiegel, 2015; Spiegel, Jain, & Albayrak, 2014) as an efficient way for converting signals to images. With this novel trend, we also benefit from the spatial dependencies of the images instead

* Corresponding author.
E-mail addresses: m_shahverdy@cmps2.iust.ac.ir (M. Shahverdy), mahfathy@iust.ac.ir (M. Fathy), rberangi@iust.ac.ir (R. Berangi), sabokro@ipm.ir (M. Sabokrou).

https://doi.org/10.1016/j.eswa.2020.113240
0957-4174/

of temporal dependencies of the driving signals, which leads us to an approach with high efficiency.

In this paper, we propose a novel and efficient method for driver behavior classification. We divide the driver behaviours into five classes: (1) safe or normal, (2) aggressive, (3) distracted, (4) drowsy, and (5) drunk driving. We gather vehicle data, including acceleration, gravity, RPM, speed, and throttle (the amount of accelerator pedal is pushed). Then, we consider overlapped time windows and use the recurrence plot technique to convert the data to the image. Finally, we use a CNN to classify the driver behavior into the five types and can alert the driver or other vehicles via wireless communication technology in VANETs.

The main contributions of this paper are: (1) we propose a method based on CNN for detecting the driver behaviors from vehicle movement patterns instead of driver's face monitoring. (2) We classify the driving behavior into the five types: normal, aggressive, distracted, drowsy and drunk driving. (3) We convert the temporal dependencies of driving signals to spatial dependencies. Experimental results show that this converting leads to a low computational power approach with high efficiency.

The rest of this paper is organized as follows. We briefly introduce the previous works in field of driver behaviors detection and machine learning techniques in Section 2. We explain the proposed approach for detecting and classifying the driver behaviors in Section 3. Experimental results and discussion are presented in Section 4. Finally, we present the conclusion and future work in Section 5.

2. Related work

2.1. Overview of driver behavior

In the literature, driving behavior is defined as the different habits, manners and actions of a driver while driving, which is classified into five styles: normal or safe driving, aggressive driving, distracted driving, drowsy driving, and drunk driving (Meiring & Myburgh, 2015). Some researchers (Al-Sultan et al., 2013; Imamura, Yamashita, Zhang, bin OTHMAN, & Miyake, 2008), define the safe driving behavior as the common daily behaviors of a specific driver, whereas an abnormal driving behavior is defined as the scarce behaviors of a specific driver whenever impressed by physical or mental factors. They have considered this problem as a one-class classification problem (Sabokrou, Khalooei, Fathy, & Adeli, 2018; Sabokrou et al., 2018). Such definition for driver behaviour is imprecise because a driver might be addicted to abnormal driving behavior and majority of his driving behaviors include anomaly behaviors. We have investigated the properties of driver behavior to distinguish different driver behavior styles and the results are summarized as follows.

2.1.1. Aggressive driving style

The impatient activities of a driver when he tries to minimize travel time. It includes tailgating, abnormal and immediate changes in vehicle speed, inappropriate keeping of vehicle lateral position, hazardous lane change, and fast acceleration and deceleration takeoff or braking (Hong, Margines, & Dey, 2014; Meiring & Myburgh, 2015).

2.1.2. Distracted driving style

Transient inattention of a driver to the task of driving and its necessary activities might make a distracted driving pattern, which usually followed with a rapid driver reaction to correct vehicle position. An aggressive driving style has a periodic pattern of misbehavior, while a distracted driving style has an instantaneous and irregular nature (Meiring & Myburgh, 2015). In Dong, Hu, Uchimura, and Murayama (2011), a diversion of attention away from critical activities, which is necessary for safe driving, toward a competing activity is defined as distracted driving. Resources of driver distraction include objects, persons or events inside or outside the car, eating or drinking, and using cell phones or other technologies in vehicle (Meiring & Myburgh, 2015).

2.1.3. Drowsy driving style

It is related to behaviors of a driver, when he/she is exhausted and try to resist against sleep. Objective signs of a fatigued driver include repeated yawning, difficulty keeping eyes open slower reaction and responses, lazy steering, vehicle wobbling in the road, rare use of brake, slow change in acceleration or gear, and moving slower than the speed limitation (Meiring & Myburgh, 2015).

2.1.4. Drunk driving style

The driver behavior, when he/she is influenced by alcohol. Drunkenness by alcohol reduces the concentration and leading to a risky behavior. Some measurable properties of drunk driving style include inappropriate keeping of vehicle lateral position, abrupt acceleration, and unsafe lane change. Unlike aggressive driving style, in a drunk driving style due to the influence of alcohol, driver's performance and driver's awareness of danger reduce significantly (Al-Sultan et al., 2013; Meiring & Myburgh, 2015).

2.1.5. Safe driving style

It is also known as normal driving or typical driving and related to the driver behavior, when he/she avoids the risky reactions and activities (Imamura et al., 2008; Meiring & Myburgh, 2015). A driving without risky activities and mentioned characteristics in careless driving, aggressive driving, drowsy driving, and drunk driving is classified in safe driving style (Meiring & Myburgh, 2015). In Miyaji, Danno, and Oguri (2008), the authors define the safe driving style, when the driver properly concentrates on driving. A safe driver should avoid tailgating, fast change in speed or acceleration, inappropriate keeping vehicle lateral position, unsafe lane change, inattention to the driving activity, and driving while fatigued or drunk.

2.2. Categorization of driver behavior detection methods based on feature type

Monitoring the driver behavior is a common field in ITS and human factors studies. Several methods have been introduced in the literature to explore human factors in the driving activity, which are categorized into two trends based on features types: (1) methods based on non-visual features and, (2) methods based on visual features. The earlier methods are based on non-visual signals of a vehicle such as acceleration, speed, and frequency use of brake (Al-Sultan et al., 2013; Carmona et al., 2015; Carmona, de Miguel, Martin, Garcia, & de la Escalera, 2016; Dai et al., 2010; Forsman, Vila, Short, Mott, & Van Dongen, 2013; Mitrovic, 2005; Tango & Botta, 2013; Van Ly, Martin, & Trivedi, 2013; Yu et al., 2017; Zhu, Liu, Zhao, Chen, & Deng, 2018) or driver (*i.e.*, Electrocardiography (ECG) and Photoplethysmography (PPG)) (Hashemi, Saba, & Resalat, 2014; Kumar, Raju, & Kumar, 2012; Li, Zhang, & Yang, 2010; Lin et al., 2010; 2005), whereas the second one uses computer vision methods to recognize some abnormal behaviors of a driver such as distraction and drowsiness via monitoring the driver's eye, head, mouth, and hand status (Abtahi, Hariri, & Shirmohammadi, 2011; Abtahi, Shirmohammadi, Hariri, Laroche, & Martel, 2013; Bhandari, Durge, Bidwai, & Aware, 2014; Cyganek & Gruszczyński, 2014; Danisman et al., 2010; Galarza et al., 2018; Hariri, Abtahi, Shirmohammadi, & Martel, 2012; Jo et al., 2014; Sabet et al., 2012; Saradadevi & Bajaj, 2008; Sun, Li, Song, & Jin, 2014). The main previous researches on driver behaviour classification task are briefly explained in the following.

2.2.1. Methods based on non-visual features

A fine-grained driving monitoring approach is suggested in Yu et al. (2017), which recognizes specific types of unusual driving behaviors such as turning with a wide radius, sudden braking, weaving, swerving, fast u-turn, and side-slipping. They gathered the orientation and acceleration of a vehicle using sensors of a smartphone in a real driving environment, and then driving behavior is detected using Neural Networks (NNs) and SVM as two machine learning algorithms.

A nonintrusive technique for detecting the visual distraction of a driver is suggested in Tango and Botta (2013), which instead of driver's eye tracking it uses vehicle signals (speed, time to collision, time to lane crossing, steering angle, lateral position, position of the accelerator pedal, position of the brake pedal). Data are gathered using a driving simulator while the participants are distracted by a secondary activity. During the driving phase, to distract the drivers, each driver is asked to complete 16 secondary tasks, with each one lasting 3 min. They utilize static and dynamic NNs, adaptive neuro fuzzy inference systems, and SVM to detect the driver distraction.

Hidden Markov model (HMM) is a stochastic tool for work with time series data, which is widely exploited for driver behavior detection (Iversen, Møller, Morales, & Madsen, 2017). In Mitrovic (2005), a method for the driving events recognition based on HMM is proposed. The system uses lateral and longitudinal acceleration and speed of a vehicle in a usual road to detect driving events.

A tool for driver behavior analysis based on data fusion is proposed in Carmona et al. (2015), which uses GPS, Inertial Measurement Unit (IMU), and in-vehicle sensors for collecting data. The maximum, mean, and standard deviation of collected data are compared with standard researches on human factors to detect the aggressive behavior of the driver.

Van Ly et al. (2013) uses the inertial sensors of vehicles to classify the driving styles. The system decreases dangerous maneuvers of the car by providing appropriate feedback to the driver. They use unsupervised algorithm (*i.e*., k-mean) and supervised algorithm (*i.e*., SVM) for detecting the driver behaviors.

Carmona et al. (2016) has suggested a tool for analysis of driver behavior based on Gaussian Mixture Model (GMM). In this work, required data are gathered from the IMU, GPS, and in-vehicle sensors using Controller Area Network bus (CAN bus). They have evaluated their proposed approach in real traffic to detect aggressive driving, which their method got the success rate of 92.65%.

Advanced driver assistance systems (ADAS) help to the driver in curve warning, curve speed control (Bosetti, Da Lio, & Saroldi, 2015), driver drowsiness identification and lane departure accidents prevention (Saito, Itoh, & Inagaki, 2016), collision avoidance (Schnelle, Wang, Su, & Jagacinski, 2016), driver intention recognition (Bengler et al., 2014), driver steering behavior analysis (You, Lu, & Tsiotras, 2017), and correct speed and headway maintenance (Bertolazzi, Biral, Da Lio, Saroldi, & Tango, 2010). In González et al. (2014), an ADAS is proposed to detect aggressive behavior of the driver by monitoring acceleration and speed of the vehicle. They model the behavior of a driver as a linear filter on the signals by using mean value, dynamic range, probability distribution functions (PDFs), and standard deviation. The proposed approach is validated in real conditions involving five types of road and ten drivers. Finally, GMM is used to detect aggressive behavior of a driver.

In Dai et al. (2010), a system for identifing and alerting drunk driving is presented. They focused on two risky behaviors of a drunk driver, including inappropriate control of speed and unfit maintenance of lane position. They use the accelerometer sensor of a smartphone to gather the data and compare them with drunk driving patterns.

Al-Sultan et al. (2013) have suggested a context-aware system in VANETs that detects abnormal behaviors of a driver and warns other vehicles to avoid traffic accidents. The system gathers the data from the driver, the vehicle, and the environment. Finally, it uses the dynamic Bayesian networks (DBNs) to deduce four types of driving behaviors, which include reckless, normal, drunk, and fatigue.

In Hansen, Busso, Zheng, and Sathyanarayana (2017), researchers have reviewed driver behavior models for distraction detection, which include some projects in UTDrive (i.e., a research platform for driver behavior modeling and In-Vehicle safety systems (UTDrive, 2015)). Their work includes detecting the driver behavior and distraction, glance behavior and visual tracking, maneuver detection, distraction analysis, a mobile platform for in-vehicle data collection and human-machine interface. They suggest two approaches for modeling the driver distraction: one approach is to extract the driving context and assess risky events. The second one is to monitor drivers' physical or glance behavior and evaluates their visual attention. Despite the valuable results of this paper, they only concentrate on driving distractions and neglect other driver behaviors.

Wang, Pei, Li, and Yao (2017) have proposed a method based on vehicle speed time series for recognizing the risky driving behaviors. At first, they have made a tuple of speed change (value and duration) for each driver. After that, a SVM classifier is exploited for identifying the risky drivers. This method is easy to apply due to the availability of speed parameter for all cars. They have achieved the classification accuracy of 95%. However, they have just focused on only one risky behavior and neglected the other unsafe behaviors such as drowsiness and distraction. Furthermore, they have ignored some risky behaviors such as driving with constant speed and risky steering.

Researchers in Liu, Taniguchi, Tanaka, Takenaka, and Bando (2017), have proposed a method for visualizing the driving behavior, which detects driver behavior patterns in continuous driving behavior data. They have used several sensors to collect the vehicle acceleration, velocity, and steering angle. Then, the hidden features for visualization of driving behavior are extracted using deep sparse autoencoder (DSAE). Their proposed method can detect the different behaviors of a driver, including high speed forward, turning left or right quickly, and accelerating forward. Their unreal-time method first collects all the driving data then infers different driving behavior patterns in all the driving time. Their method is unable to identify and alert the driving behavior in real-time and just focus on offline reporting.

Škrjanc et al. (2018) have proposed an evolving cloud-based method based on the car signals for driver action recognition. The exploited signal by them are: speed, RPM, steering wheel, gas pedal, brake pedal, clutch pedal, and gear. The data are preprocessed and used to form clouds called atomic actions. The combination of atomic actions forms complex sequences called tasks. Also, a sequences of different tasks form a complex structure called manoeuvre. Their evolving cloud-based algorithm can recognize the complex action of a driver, including overtaking, stopping, stopping at traffic light, and keeping safety distance manoeuver.

2.2.2. Methods based on visual features

Some researchers monitor the driver's eye and analyze the eye condition (*i.e*., eye open/close duration and eye blink rate) to detect the drowsy driving (Cyganek & Gruszczyński, 2014; Danisman et al., 2010; Jo et al., 2014; Sabet et al., 2012; Sun et al., 2014) whereas some others monitor the driver's mouse an yawing to detect the drowsy driving (Abtahi et al., 2011; Abtahi et al., 2013; Bhandari et al., 2014; Hariri et al., 2012; Saradadevi & Bajaj, 2008).

A monitoring system was proposed in Galarza et al. (2018) to detect and alert the driver about the drowsiness driving. Camera

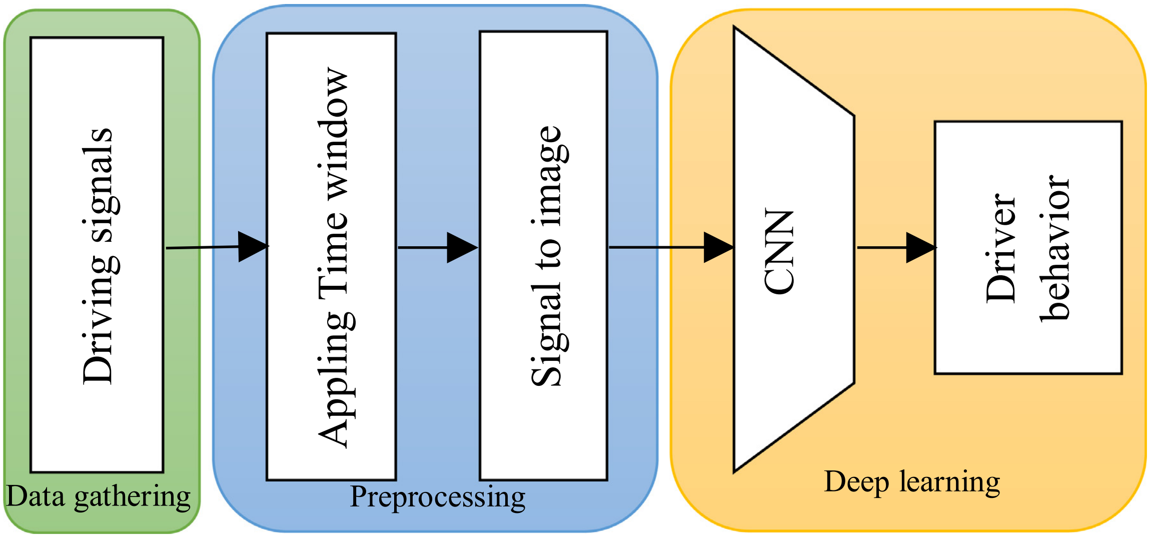

Fig. 1. The proposed approach for the driver behavior detection system. The collected data, including acceleration, gravity, speed, RPM, and throttle are windowed. Then, the signals are converted to images by recurrence plot technique. Generated images will be given to a CNN to classify them to the five driving styles.

of a mobile monitors head pose, eye conditions, and yawing. Their method gets 93.37% accuracy in drowsiness detection in the natural lighting and no matter the driver use of cap or glasses.

In Yan, Teng, Smith, and Zhang (2016), a method based on visual features of drivers was proposed, which can detect the driver distraction using CNN. The method extracts skin-like regions by GMM then pass them to an R*CNN to generate action labels. It was tested on Southeast University Driving-posture dataset, which includes driver's image while phone call, eating, operating the shift gear, playing phone, or smoking. The proposed method achieves an accuracy of 97.76%.

In Lee, Yoon, Song, and Park (2018), a method was proposed for detecting aggressive driving by CNN. Input images include driver's face, which obtains using near-infrared (NIR) light and thermal camera sensors. The proposed method achieves an accuracy of 99.95%.

Streiffer, Raghavendra, Benson, and Srivatsa (2017) proposed a framework, DarNet, which can detect distracted driving behavior. DarNet includes two main components: (1) a data collection system and (2) an analysis engine. They collect image from an inward-facing camera and IMU from a smartphone. The images and IMU signals are applied to a CNN and Recurrent Neural Network (RNN), respectively. Then, the outputs of two networks are combined by a bayesian network to classify the driver behaviors. DarNet achieves an accuracy of 87.02%.

Baheti, Gajre, and Talbar (2018) suggested a CNN-based system, which detects the distracted driver and the cause of distraction. Distracted driver behaviors include texting, talking on the phone, adjusting radio, drinking, hair and makeup, talking to passengers, and reaching behind. They use the modified VGG-16 architecture and different regularization techniques to improve the performance. Their method achieves an accuracy of 96.31%. Also, they proposed a modified version of the network, which achieves an accuracy of 95.54% with 15M parameters (that is 140M in original VGG-16).

In Jo et al. (2014), a method for classification of eye state was developed that extracts features from both eyes. The system makes a model of normal (non-drowsy) driver-blinking patterns in different texture and eye shape then detects the eye state as closed or open.

Driver behavior detection techniques, which are based on visual features, are too sensitive to light conditions. Despite modern cameras, severe change in light intensity decreases the accuracy of these techniques. Furthermore, the driver might be upset with intrusive techniques, which use ECG, PPG, or camera. Image processing needs high computational resource, which is not appropriate for the real-time applications and embedded systems in vehicles. Instead, driver behavior detection techniques, which are based on non-visual features needs low computational resource but their accuracy are low. Hence, we suggest a technique based on non-visual features of the driver, which provide high accuracy and efficiency.

2.3. Machine learning techniques in driver behavior detection

The popular algorithms in driver behavior detection and driving style studies are summarized In Meiring and Myburgh (2015). Artificial Neural Networks (ANNs) are applied in distraction detection, drowsiness detection, and steering behavior prediction, which are based on computer vision techniques (Daza et al., 2011; Tango & Botta, 2013). Clustering techniques (*i.e.*, K-means) are used in driving styles distinction (Castignani, Derrmann, Frank, & Engel, 2015). HMMs are widely used for assessment of driver performance and driver behavior (Fu, Wang, & Zhao, 2016; Gadepally, Krishnamurthy, & Ozguner, 2014; Gadepally, Kurt, Krishnamurthy, & Özgüner, 2011; Oliver & Pentland, 2000), driving manoeuver detection (Boyraz, Acar, & Kerr, 2007; Mitrovic, 2005; Oliver & Pentland, 2000), and driver distraction detection (Takeda et al., 2011). GMMs are applied in detection of driver distraction and driving manoeuver (Sathyanarayana, Boyraz, & Hansen, 2008; Takeda et al., 2011) and road condition monitoring (Almazán, Bergasa, Yebes, Barea, & Arroyo, 2013). SVMs are used in driving style classification (Wang, Xi, Chong, & Li, 2017), driving event detection (Sathyanarayana, Sadjadi, & Hansen, 2012; Van Ly et al., 2013), drowsiness detection (Li, Jin, Jiang, Xian, & Gao, 2015), and vehicle state assessment (Bhoraskar, Vankadhara, Raman, & Kulkarni, 2012). Deep learning are used for driver stress level classification In Rastgoo, Nakisa, Maire, Rakotonirainy, and Chandran (2019) and driver activities classification In Danisman et al. (2010); Galarza et al. (2018); Jo et al. (2014); Sabet et al. (2012).

Recurrent Neural Networks as a subset of deep learning methods are used for driver behavior profiling in Carvalho et al. (2017) so that three axes accelerations are applied as three inputs into Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) and these two types of RNN clas-

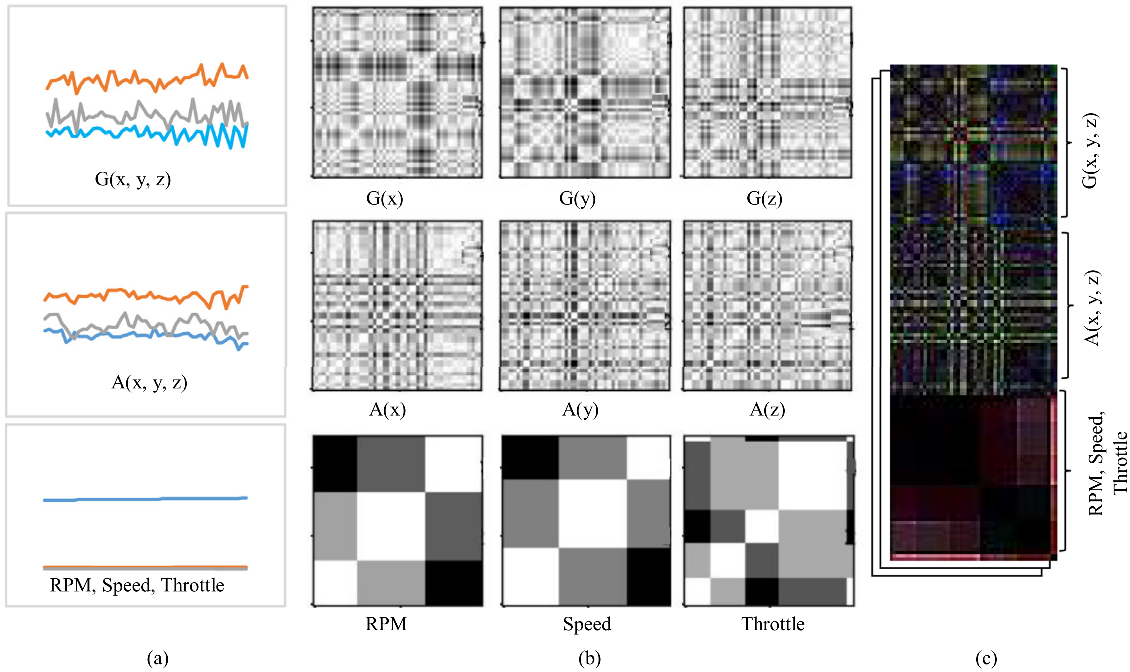

Fig. 2. (a) Three axes gravity signals, three axes acceleration signals, RPM, speed and throttle, which are collected in data gathering phase. (b) Nine driving signals are converted into nine grayscale 50 × 50 pixels images. (c) Each of triple grayscale images are considered as a RGB image with three colored channels.

sify the driving event types. In Jain, Singh, Koppula, Soh, and Saxena (2016), authors use the RNN method for driver activity anticipation via sensory-fusion architecture. They track the driver's face and head-pose, then they use the Fusion-RNN to combine the features from inside the vehicle (driver's face), with the features from outside the vehicle (GPS, road camera, and vehicle dynamics).

3. Proposed approach

Fig. 1 outlines the proposed approach for driver behavior detection. The acceleration and gravity are gathered in three axes from a smartphone, which is fixed in the vehicle. An On-Board Diagnostic (OBDII) adapter gathers the RPM, speed, and throttle of vehicle from Engine Control Unit (ECU).

We apply a time window on nine collected signals to distinguish the different driver behaviors. Then all windowed data are converted to the images by recurrence plot technique. It helps us to convert the temporal dependencies of driving signals to spatial dependencies. Generated images will be given to a CNN to classify them into five driving styles. We experiment various configurations of CNNs, including different filter sizes, the number of convolutional layers, and the number of filters to achieve a model with low computational cost and high efficiency.

3.1. Data gathering

In order to experiment the proposed method, we carried out a set of driving under real conditions. We chose a real scenario rather than a simulation-based scenario to avoid probable deviations from the actual behavior of drivers.

Three drivers have participated in conducting the experiments. They drove with the same vehicle under real driving conditions. Participants were asked to drive in all types of driving, including (1) normal, (2) aggressive, (3) distracted, (4) drowsy, and (5) drunk. They knew the attributes of all driving styles and imitated the different driving behaviors with doing some predesignate jobs. We gathered data, including acceleration, gravity, RPM, speed, and throttle of the vehicle during the experiment.

To analyze the vehicle movement, we need the linear acceleration of the vehicle regardless of the vehicle orientation. The linear acceleration measures the acceleration force that is applied to a vehicle on all three axes (x, y, and z), excluding the force of gravity. For calculating the linear acceleration, we need both acceleration and gravity signals in all three axes (x, y, and z). We consider the acceleration and gravity as two features and delegate them to the deep neural network for automatic feature evaluation and representation learning.

RPM is the frequency of engine rotation around a fixed axis in one minute. We use an OBDII adapter for extracting the RPM value from parameters of ECU. The throttle position sensor monitors how much throttle valve is opened, which it is associated with the amount of accelerator pedal push. This parameter is read from the ECU by OBDII adapter. The amount of RPM often changes proportional to throttle but in cases such as uphill or downhill road and especially in manual (not automatic) cars, these parameters may not change proportionally. Therefore, we consider both features for the accurate estimation of driver behavior.

Aggressive driving was emulated by a driver with aggressive activities such as high speed, tailgating, severe acceleration, and deceleration while take-off or braking. To imitate the distracted driving style, a dummy activity was used to distract the attention of the driver from the road for an interval between 1 and 2 s (Tango & Botta, 2013). For this purpose, the driver should type 20 to 40 characters as a SMS in a cell phone. The driver should repeat dummy activity after a random time between 3 and 9 s (Tango & Botta, 2013). A driver assistance helped the driver to schedule the time of dummy activities. Drowsy driving was imitated by closing eyes for a random time between 1 and 2 s, rare use of brake, lazy steering, moving at a slower speed than the speed limitation, and low change in acceleration or gear. Actually, a drowsy driver finally becomes conscious and correct the vehicle position or have an accident.We just do the first one and prevent the accident by help of another person who warns the danger and schedules time of activities. Drunk driving was imitated by high speed, abrupt acceleration, fast steering, and less use of brake. Safe driving was done,

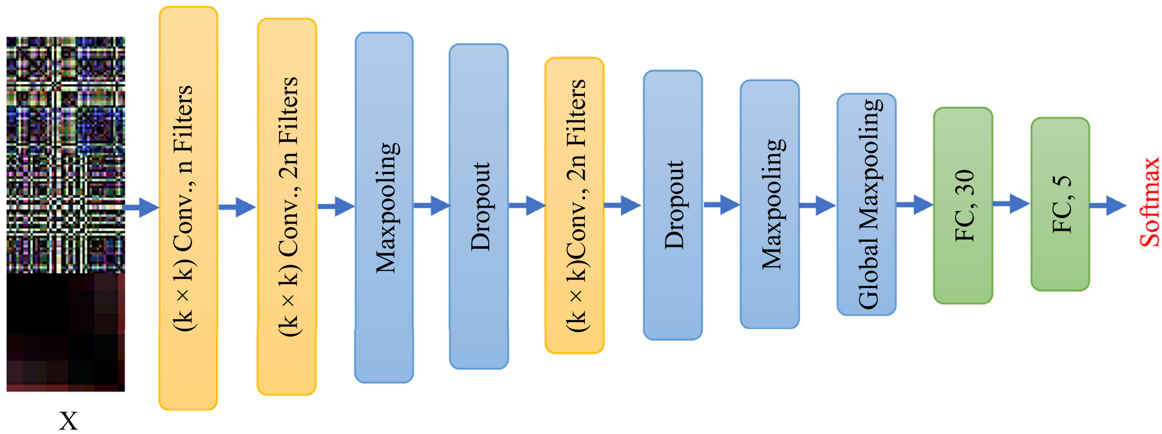

Fig. 3. CNN architecture with three convolutional layers that uses n filters of $k \times k$ in the first convolutional layer. In Section 4, we experiment this network with various k, n, and various numbers of convolutional layers to achieve a model with high efficiency. For a model with two convolutional layers, the 3rd convolutional layer and 2nd maxpooling are removed.

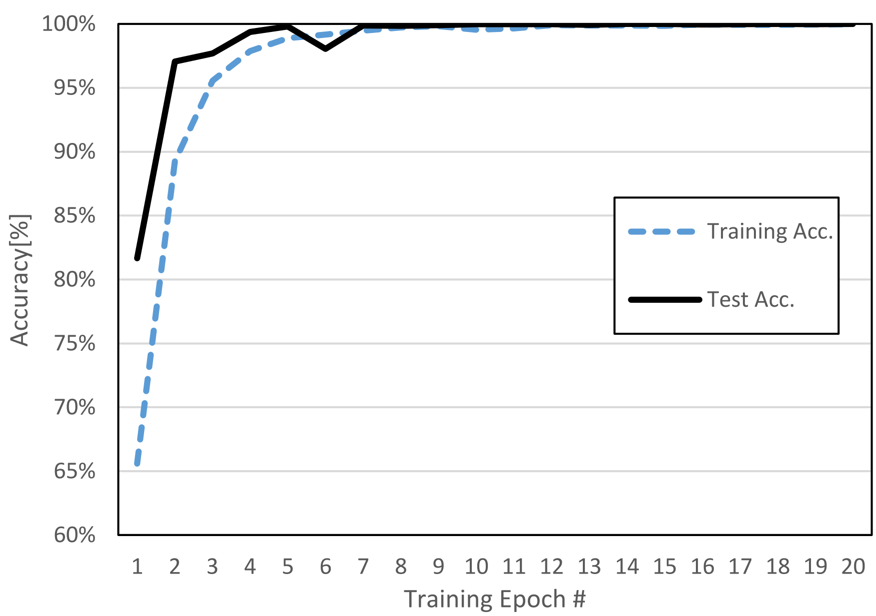

Fig. 4. Accuracy versus number of training epochs for the 7th model. The model approximately start to converge after the 4th epoch.

including safe acceleration, safe speed, safe distance to front vehicle, and safe steering.

In the literature, reaction time for expected, unexpected, and surprise situations vary greatly between about 0.7 to 3 s and the average driver reaction time is 2.3 s (McGehee, Mazzae, & Baldwin, 2000; Summala, 2000). To detect tiny fluctuation of acceleration, we set the frequency of data gathering to 10 Hz, which is much smaller than the minimum reaction time of a driver. All data of various driving styles was extensively collected in various kinds of road and traffic conditions.

3.2. Applying time window and recurrence plot

We monitor the driver behaviors in a time slot, which each slot has a label of normal, aggressive, distracted, drowsy, or drunk. In Dai et al. (2010), a 5 s time window is considered for detection of drunk driving; In Carmona et al. (2016), a 20 s time window with a 50% overlap is considered for detection of an aggressive driving; In Tango and Botta (2013), a 1.8 s time window is considered for detection of a distracted driving. Empirically, we consider a 50-ms time window with a 98% overlap to distinguish the different driver behaviors. Data are collected in time domain and has temporal dependency. The recurrence plot (RP) is used to visualize recurrent states of time series (Spiegel, 2015; Spiegel et al., 2014). We use the RP on temporal data to make images with spatial properties. The implicit mathematical of RP is as follows:

$$R^X_{i,j} = \theta(\epsilon - \|X_i - X_j\|) \quad (1)$$

where $\|.\|$ is a norm, x is a time series, and θ is the Heaviside function. The ϵ is a recurrence threshold, which affects the formation of line structures.

We use the Pyts, which is a python package for the time series transformation and recurrence plotting (Faouzi, 2018). Fig. 2 shows the transformation progress of driving signals to a colored image using the recurrence plot. Three axes gravity signals are converted to three grayscale 50 × 50 pixels images; three axes acceleration signals are converted to three grayscale 50 × 50 pixels images; RPM, speed, and throttle signals are converted to three grayscale 50 × 50 pixels images. We consider each triple grayscale images as a RGB image with three colored channels. Final images have

Table 1
Number of samples and plotted images in each class.

Driving style	Number of samples	Number of plotted images
Safe	3638	3587
Aggressive	4977	4926
Distracted	4412	4312
Drowsy	5093	5040
Drunk	3291	3241

150×50 pixels in each channel. Table 1 shows the details of the dataset, including number of samples and number of plotted images in each class.

3.3. CNN Architecture

Deep neural network automatically learns the discriminative features. Such networks need to train on a huge number of samples to provide an informative representation of them. The first layers of deep neural networks usually learn the low-level features while the last layers are responsible for fusing the low-level features to provide a high-level representation (*i.e.*, features) for its input. CNN fundamentally is a type of ANN, which is used in various deep learning applications like object detection, image classification, natural language processing, action recognition, and many more. The basic components of a CNN include convolutional layers, pooling layers, activation functions, and fully connected layers, which is basically stacked one after the other to make a network. Since 2012, due to the availability of computing power and large amount of labeled data, CNNs have been widely used. Various architectures such as ResNet (Zeiler & Fergus, 2013), AlexNet (Russakovsky et al., 2015), VGGNet (Simonyan & Zisserman, 2014), ZFNet (Zeiler & Fergus, 2013), GoogLeNet (Szegedy et al., 2015), have been used for computer vision applications.

In this paper, we apply new trends of CNN architectures to achieve the best accuracy and computational cost for driver behavior detection. A major different architecture of CNN architecture used is to avoid a very deep fully connected network, which has a lot of learned parameters. Instead, we use global max pooling at the end of CNN and a very small fully connected network. These new trends of CNN show having high efficiency, very fewer parameters and less prone to over-fitting in many applications. As the image generation by our method has structured patterns and higher spatial properties exists in different input images; compared with conventional natural images, just few convocational layers are needed for training and convergence. With considering these frameworks, however many architectures with different small numbers of convoulutional layers, number of filters and filter size are designed and compared.

We use the Leaky Rectified Linear Unit (Leaky ReLU) activation function, which due to its efficiency and faster convergence has become so popular in the past few of years. In simple ReLU due to the zero gradient, some units of the network will be inactivated, whereas Leaky ReLU gives a small and positive gradient when the unit is inactive (Xu, Wang, Chen, & Li, 2015).

All input images are 150×50 pixels in each three channels. Initial layers of the CNN include convolutional and max pooling layers act as feature extractor, while the last layer is softmax classifier, which classifies the input images into one of the five categories. Dropout is an efficient regularization technique, which randomly ignoring some neurons to reduce over-fitting (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). We use two dropout of 0.25 in convolutional layers. The network architecture is shown in Fig. 3. Applying time window and recurrence plot phase are described in the pseudo Algorithm 1 . Generating, training, and testing of different CNN models are presented in pseudo Algorithm 2 .

Algorithm 1: Pseudo algorithm for windowing and recurrence plot.

```
Input  : X^(m) = (x_1^(m) + x_2^(m) + ··· x_n^(m))
Output: Image^(k)
Window size: ws = 50
Sliding window: sw = 1 //98% overlapping.
recurrence threshold: ϵ
RecurrencePlots function: R^X_{i,j} = θ(ϵ − ‖X_i − X_j‖)
for samples X^(m) ∈ dataset do
    for features f ∈ features X^(m) do
        Image_f^(K) = R^{X_f} //R apply to each window of each
        feature.
    end
    Image^(k) = concatenate(Image_f^(K)) // concatenate the
    images to make an image for each sample.
end
```

Algorithm 2: Pseudo algorithm for CNN training and testing.

```
Input  : (X_train, Y_train) and (X_test, Y_test) // Image set is divided
         into two sets, including 75% of Image for train set
         and 25% of Image for test set
Initialize (parameters): batch_size = 32, epochs = 20, α =
0.2, drop_rate = 0.25, pool_size = (2, 2), num_classes = 5
for number of convolutional layers layerCnt ∈ {2, 3} do
    for filter size filter_size ∈ {2, 3, 5, 7} do
        for number of filters filterCnt ∈ {16, 32} do
            model = Sequential()
            model.add(Conv2D(filterCnt,
            (filter_size, filter_size)))
            model.add(LeakyReLU(α))
            model.add(Conv2D(2 ∗ filterCnt, (filter_size,
            filter_size)))
            model.add(LeakyReLU(α))
            model.add(MaxPooling2D(pool_size))
            model.add(Dropout(drop_rate))
            if layerCnt == 3 then
                model.add(Conv2D(2 ∗ filterCnt, (filter_size,
                filter_size)))
                model.add(LeakyReLU(α))
                model.add(Dropout(drop_rate))
                model.add(MaxPooling2D(pool_size))
            end
            model.add(GlobalMaxPooling2D())
            model.add(Dense(30, activation = 'relu'))
            model.add(Dense(num_classes, activation =
            'softmax'))
            model.compile(loss =
            losses.categorical_crossentropy, optimizer =
            optimizers.Adadelta())
            model.fit(X_train, Y_train, batch_size, epochs,
            validation_data = (X_test, Y_test))
            model.evaluate()
        end
    end
end
```

In Section 4, we investigate the influence of filter size, the number of convolutional layers, and the number of filters on the accuracy of the proposed model.

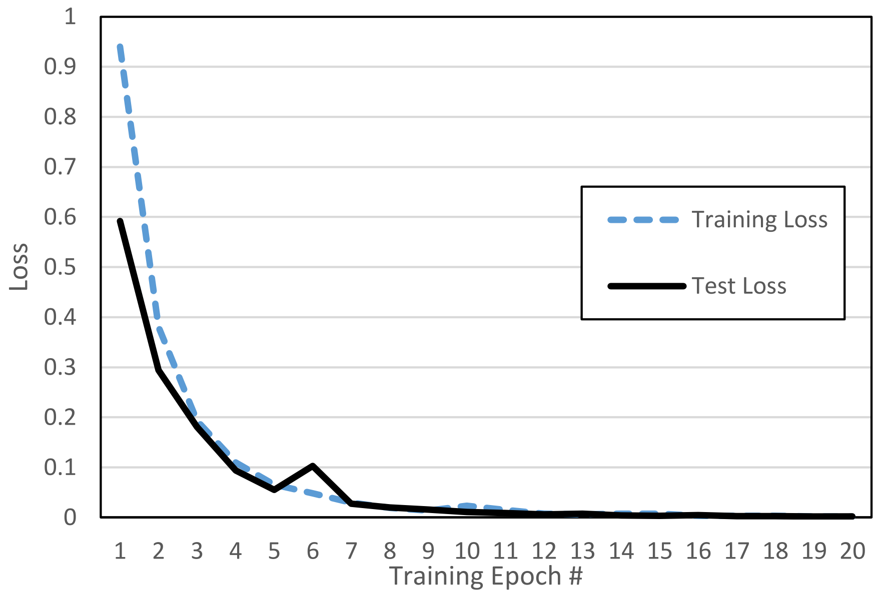

Fig. 5. Loss versus number of training epochs for the 7th model.

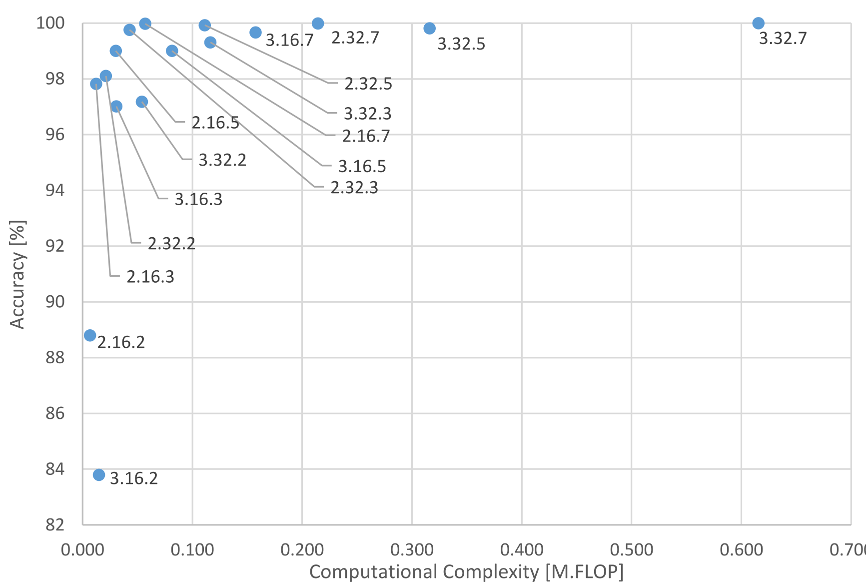

Fig. 6. Trade off between accuracy and computational complexity. Each node is a model, which is labeled as a triplex (number of convolutional layers. number of filters in the 1st layer.filter size).

4. Experimental results and discussion

In this section, we evaluate the performance of the proposed driver behavior detection system. The system detects five types of driving behaviors by multiclass classification. We use an array unison shuffle technique to randomly shuffle all samples of five classes. Then we divide the dataset into two non-overlapping sets, including 75% of the dataset for the training set and 25% of the dataset for the test set.

We experiment various configurations of CNNs, including different filter size, the number of convolutional layers, and the number of filters to achieve a simple yet efficient network. Moreover, we experiment a configuration of CNN to achieve a model with low computational cost and high efficiency, which is appropriate for embedded applications.

4.1. Experiment settings

We use the TensorFlow (Abadi et al., 2016) and keras library in python (Gulli & Pal, 2017) to implement the proposed network on a computer system with a CPU intel core i7, 32GB RAM, NVIDIA GeForce 1080Ti. The batch size and the epoch size are set to 32 and 20, respectively.

We use an adaptive learning rate method (Adadelta) as the parameter of optimizer algorithm with default hyperparameters (initial learning rate=1.0, and decay factor ρ=0.95) during the training process. The method dynamically adapts over time using only first-order data with minimal computational overhead beyond vanilla stochastic gradient descent. The method requires no manual tuning of a learning rate and appears robust to the selection of hyperparameters, different model architecture, noisy gradient information, and various data modalities (Zeiler, 2012).

Table 2
Various models with different number of convolutional layers, number of filters in the 1st layer, and filter size. The number of filters in the 2nd and 3rd convolutional layers is twice the number of filters in the 1st convolutional layer. Number of parameters, accuracy, and computational complexity are listed for each model.

Model#	#of Conv. Layers	# of Filters in the 1st Layer	FilterSize	#of Params [K]	Computational Complexity [M.FLOP]	Accuracy[%]
1	2	16	2 × 2	3.433	0.007	88.79
2	2	16	3 × 3	6.233	0.012	97.82
3	2	16	5 × 5	15.193	0.030	99.01
4	2	16	7 × 7	28.633	0.057	99.98
5	2	32	2 × 2	10.777	0.021	98.10
6	2	32	3 × 3	21.497	0.043	99.76
7	2	32	5 × 5	55.801	0.111	99.93
8	2	32	7 × 7	107.257	0.214	99.99
9	3	16	2 × 2	7.561	0.015	83.80
10	3	16	3 × 3	15.481	0.031	97.01
11	3	16	5 × 5	40.825	0.081	99.01
12	3	16	7 × 7	78.841	0.157	99.67
13	3	32	2 × 2	27.225	0.054	97.18
14	3	32	3 × 3	58.425	0.116	99.31
15	3	32	5 × 5	158.265	0.316	99.81
16	3	32	7 × 7	308.025	0.616	99.99

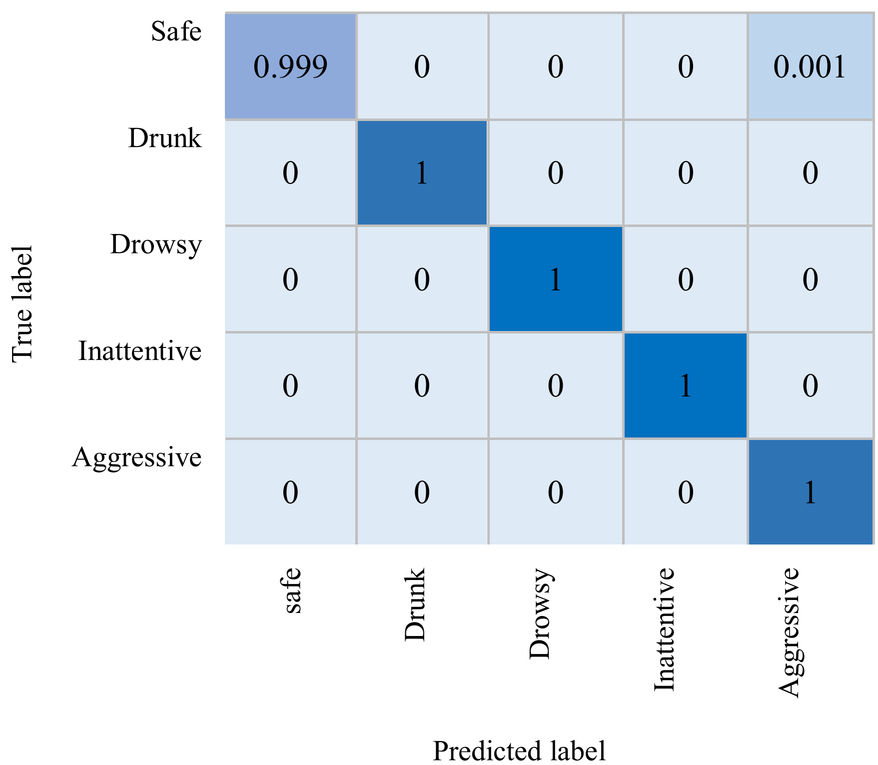

Fig. 7. Confusion matrix of driver behavior classification for the CNN model with 2 convolutional layers, 16 filters in first layer, and filter size of 7 × 7.

4.2. Evaluation of driver behavior detection system

We evaluate the performance of our method in respect to its accuracy and computational complexity. The accuracy measure is calculated based on Eq. (2).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Where True Positives (TP) and True Negatives (TN) are the positive and negative instances respectively, which correctly predicted. False Positives (FP) and False Negative (FN) are the positive and negative instances respectively, which incorrectly predicted.

The computational complexity of our method is reported based on Million Floating Point Operations (MFLOP). Floating point operations include add and multiplication operations in the network.

The accuracy and computational complexity of different networks are illustrated in Table 2. With respect to trade off between accuracy and computational complexity in Fig. 6, the 4th model (with 2 convolutional layers, 16 filters in the first layer, and filter size of 7 × 7) is the best model and is considered as our reference model. The test accuracy of the reference model is equal to 99.98% and the test loss of the experiment is equal to 0.48%. Regardless of computational complexity, some networks like 8th and 16th model in Table 2, which both of them have 32 filters with the size of 7 × 7, achieve higher accuracy near 99.999%.

We infer from Table 2 that filter size is the most effective parameter on accuracy and number of filters is the second most effective parameter on accuracy. As mentioned before due to special structure of input images obtained by our method, the number of convolutional layers has an insignificant influence on the accuracy of the models, therefore very deep networks have no necessarily presented the best results.

The accuracy and loss of the 7th model versus number of training epochs are illustrated in Fig. 4 and Fig. 5, respectively. The model starts to converge after the 4th epoch.

Recently, driver behavior detection systems are widely attended by researchers (see a summary of them in Section 2). The idea of using machine-learning methods to detect driver behavior is not completely novel. In our point of view, Carvalho et al. (2017) and Al-Sultan et al. (2013) are the most closest works to ours. These works are unable to classify the driver behaviors into five styles as we have developed here. Since the goal of these researches is different, comparing our proposed approach with mentioned methods is not feasible.

In Carvalho et al. (2017), the authors use the vehicle acceleration in three axes as the input of LSTM, GRU, and Simple RNN to detect the aggressive driving behavior, which the best result for the GRU and LSTM is approximately 95% and for the SimpleRNN is approximately 70%. However their system is unable to classify the driving behavior into five styles as we have developed here. So, there is no basis to compare our method with their approach, which is also based on deep learning but with totally different architecture for just detecting aggressive driving style. Nevertheless, we examine our method with the similar features, which are used in Carvalho et al. (2017). To this end, we use just vehicle acceleration in three axes as an input of the system. The accelerations of the vehicle are converted to the images of 50×50 pixels by the recurrence plot and are applied to a CNN with 2 convolutional layers, 32 filters in the first layer, and filter size of 5×5. The accuracy of our method just with the acceleration feature is 99.95%.

To highlight the performance of our method, confusion matrix of the proposed driver behavior detection system for the reference model with 2 convolutional layers, 16 filters in first layer, and filter size of 7×7 is illustrated in Fig. 7. All instances of drowsy, aggressive, inattentive, and drunk driving behaviors were classified correctly but 0.1% of safe behavior instances incorrectly classified as aggressive behavior.

4.3. Embedded approach

The goal of designing a driver behavior detection system is to use the system as an embedded device in the vehicle, therefore the model should have low computational cost to efficiently work on embedded systems. The 6th model in Table 2 with 2 convolutional layers, 32 filters in first layer, and filter size of 3×3 is the most appropriate model for embedded applications because it has 21.5 k parameters and computational complexity of 0.043 MFLOP that is low computational cost while it has high accuracy of 99.76%. The proposed approach presented in this paper, which its major contribution is to convert the time dependency of input signals to images with spatial dependency has guided us to a very small system and new trends of CNNs with high accuracy and low computational power.

5. Conclusion

In this paper, we have proposed a method for analyzing driver behavior based on vehicle signals during driving, instead of monitoring the driver's visual features. We have collected driving data in different roads and different driver behaviors. Recurrence plot technique is used to convert the driving signals, including acceleration, gravity, throttle, speed, and RPM to the images and then a CNN is used to classify the image to five types of driving styles, including normal, aggressive, distracted, drowsy, and drunk driving. We have profited from the spatial dependencies in images rather than temporal dependencies of driving signals to achieve an approach with low computational cost and high efficiency. The evaluation results demonstrate the efficiency of our driver behavior detection system.

Finally, the proposed system can alert the danger to the police, the driver, and other vehicles. Future works include optimizing the size of time windows for higher performance; investigating the other learning algorithm such as online learning; also, we are interested in estimating the probability of accident risk collaboratively by combining drivers' behavior and environment status in vehicular networking.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Credit authorship contribution statement

Mohammad Shahverdy: Conceptualization, Methodology, Software, Validation, Investigation, Resources, Data curation, Writing - original draft, Writing - review & editing, Visualization. **Mahmood Fathy:** Conceptualization, Supervision, Writing - review & editing, Project administration. **Reza Berangi:** Supervision. **Mohammad Sabokrou:** Conceptualization, Formal analysis, Writing - review & editing.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., … Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In 12th Symposium on operating systems design and implementation: *16* (pp. 265–283).

Abtahi, S., Hariri, B., & Shirmohammadi, S. (2011). Driver drowsiness monitoring based on yawning detection. In Proceedings, IEEE international instrumentation and measurement technology conference (pp. 1–4).

Abtahi, S., Shirmohammadi, S., Hariri, B., Laroche, D., & Martel, L. (2013). A yawning measurement method using embedded smart cameras. In Proceedings, IEEE international instrumentation and measurement technology conference (pp. 1605–1608).

Al-Sultan, S., Al-Bayatti, A. H., & Zedan, H. (2013). Context-aware driver behavior detection system in intelligent transportation systems. IEEE Transactions on Vehicular Technology, *62*(9), 4264–4275.

Almazán, J., Bergasa, L. M., Yebes, J. J., Barea, R., & Arroyo, R. (2013). Full auto-calibration of a smartphone on board a vehicle using IMU and GPS embedded sensors. In Proceedings IEEE intelligent vehicles symposium (iv) (pp. 1374–1380).

Baheti, B., Gajre, S., & Talbar, S. (2018). Detection of distracted driver using convolutional neural network. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 1032–1038).

Bengler, K., Dietmayer, K., Farber, B., Maurer, M., Stiller, C., & Winner, H. (2014). Three decades of driver assistance systems: Review and future perspectives. IEEE Intelligent Transportation Systems Magazine, *6*(4), 6–22.

Bertolazzi, E., Biral, F., Da Lio, M., Saroldi, A., & Tango, F. (2010). Supporting drivers in keeping safe speed and safe distance: The saspence subproject within the european framework programme 6 integrating project prevent. IEEE Transactions on Intelligent Transportation Systems, *11*(3), 525–538.

Bhandari, G., Durge, A., Bidwai, A., & Aware, U. (2014). Yawning analysis for driver drowsiness detection. International Journal of Engineering Research and Technology, *3*(2), 502–505.

Bhoraskar, R., Vankadhara, N., Raman, B., & Kulkarni, P. (2012). Wolverine: Traffic and road condition estimation using smartphone sensors. In Proceedings fourth international conference on communication systems and network (pp. 1–6).

Bosetti, P., Da Lio, M., & Saroldi, A. (2015). On curve negotiation: From driver support to automation. IEEE Transactions on Intelligent Transportation Systems, *16*(4), 2082–2093.

Boyraz, P., Acar, M., & Kerr, D. (2007). Signal modelling and hidden markov models for driving manoeuvre recognition and driver fault diagnosis in an urban road scenario. In Proceedings of IEEE intelligent vehicles symposium (pp. 987–992).

Carmona, J., García, F., Martín, D., Escalera, A., & Armingol, J. (2015). Data fusion for driver behaviour analysis. *Sensors*, *15*(10), 25968–25991.

Carmona, J., de Miguel, M. A., Martin, D., Garcia, F., & de la Escalera, A. (2016). Embedded system for driver behavior analysis based on GMM. In Proceedings of IEEE intelligent vehicles symposium (iv) (pp. 61–65).

Carvalho, E., Ferreira, B. V., Ferreira, J., de Souza, C., Carvalho, H. V., Suhara, Y., … Pessin, G. (2017). Exploiting the use of recurrent neural networks for driver behavior profiling. In Proceedings of international joint conference on neural networks (pp. 3016–3021).

Castignani, G., Derrmann, T., Frank, R., & Engel, T. (2015). Driver behavior profiling using smartphones: A low-cost platform for driver monitoring. IEEE Intelligent Transportation Systems Magazine, *7*(1), 91–102.

Cyganek, B., & Gruszczyński, S. (2014). Hybrid computer vision system for drivers' eye recognition and fatigue monitoring. *Neurocomputing, 126*, 78–94.

Dai, J., Teng, J., Bai, X., Shen, Z., & Xuan, D. (2010). Mobile phone based drunk driving detection. In Proceedings of 4th international conference on pervasive computing technologies for healthcare (pp. 1–8).

Danisman, T., Bilasco, I. M., Djeraba, C., & Ihaddadene, N. (2010). Drowsy driver detection system using eye blink patterns. In Proceedings of IEEE international conference on machine and web intelligence (pp. 230–233).

Daza, I. G., Hernández, N., Bergasa, L. M., Parra, I., Yebes, J. J., Gavilán, M., … Sotelo, M. (2011). Drowsiness monitoring based on driver and driving data fusion. In Proceedings of 14th International IEEE Conference on Intelligent Transportation Systems (pp. 1199–1204).

Dong, Y., Hu, Z., Uchimura, K., & Murayama, N. (2011). Driver inattention monitoring system for intelligent vehicles: A review. IEEE Transactions on Intelligent Transportation Systems, *12*(2), 596–614.

Faouzi, J. (2018). *pyts Documentation: Python package for time series transformation and classification*.

FARS (2017). Fatality analysis reporting system (fars) encyclopedia. https://www-fars.nhtsa.dot.gov/Main/index.aspx.

Forsman, P. M., Vila, B. J., Short, R. A., Mott, C. G., & Van Dongen, H. P. (2013). Efficient driver drowsiness detection at moderate levels of drowsiness. *Accident Analysis & Prevention, 50*, 341–350.

Fu, R., Wang, H., & Zhao, W. (2016). Dynamic driver fatigue detection using hidden Markov model in real driving condition. Expert Systems With Applications, *63*, 397–411.

Gadepally, V., Krishnamurthy, A., & Ozguner, U. (2014). A framework for estimating driver decisions near intersections. IEEE Transactions on Intelligent Transportation Systems, *15*(2), 637–646.

Gadepally, V., Kurt, A., Krishnamurthy, A., & Özgüner, Ü. (2011). Driver/vehicle state estimation and detection. In Proceedings of 14th International IEEE conference on intelligent transportation systems (pp. 582–587).

Galarza, E. E., Egas, F. D., Silva, F. M., Velasco, P. M., & Galarza, E. D. (2018). Real time driver drowsiness detection based on drivers face image behavior using a system of human computer interaction implemented in a smartphone. In International conference on information theoretic security (pp. 563–572). Springer.

González, A. B. R., Wilby, M. R., Díaz, J. J. V., & Ávila, C. S. (2014). Modeling and detecting aggressiveness from driving signals. IEEE Transactions on Intelligent Transportation Systems, *15*(4), 1419–1428.

Gulli, A., & Pal, S. (2017). *Deep learning with keras*. Packt Publishing Ltd.

Hansen, J. H., Busso, C., Zheng, Y., & Sathyanarayana, A. (2017). Driver modeling for detection and assessment of driver distraction: Examples from the utdrive test bed. IEEE Signal Processing Magzine, *34*(4), 130–142.

Hariri, B., Abtahi, S., Shirmohammadi, S., & Martel, L. (2012). A yawning measurement method to detect driver drowsiness. *Technical Papers*.

Hashemi, A., Saba, V., & Resalat, S. N. (2014). Real time driver's drowsiness detection by processing the eeg signals stimulated with external flickering light. *Basic and Clinical Neuroscience, 5*(1), 22.

Hong, J.-H., Margines, B., & Dey, A. K. (2014). A smartphone-based sensing platform to model aggressive driving behaviors. In Proceedings of the SIGCHI conference on human factors in computing systems (pp. 4047–4056). ACM.

Imamura, T., Yamashita, H., Zhang, Z., bin OTHMAN, M. R., & Miyake, T. (2008). A study of classification for driver conditions using driving behaviors. In Proceedings of IEEE international conference on systems, man and cybernetics (pp. 1506–1511).

Iversen, E. B., Møller, J. K., Morales, J. M., & Madsen, H. (2017). Inhomogeneous Markov models for describing driving patterns. IEEE Transactions on Smart Grid, *8*(2), 581–588.

Jain, A., Singh, A., Koppula, H. S., Soh, S., & Saxena, A. (2016). Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In Proceedings of IEEE international conference on robotics and automation (pp. 3118–3125).

Jo, J., Lee, S. J., Park, K. R., Kim, I.-J., & Kim, J. (2014). Detecting driver drowsiness using feature-level fusion and user-specific classification. Expert Systems with Applications, *41*(4), 1139–1152.

Kumar, G. R., Raju, S. V. P., & Kumar, D. S. (2012). Classification of eeg signals for drowsiness detection in brain and computer interface. *GESJ: Computer Science and Telecommunications, 4*(36).

Lee, B.-G., & Chung, W.-Y. (2012). A smartphone-based driver safety monitoring system using data fusion. *Sensors, 12*(12), 17536–17552.

Lee, K., Yoon, H., Song, J., & Park, K. (2018). Convolutional neural network-based classification of drivers emotion during aggressive and smooth driving using multi-modal camera sensors. *Sensors, 18*(4), 957.

Li, K., Jin, L., Jiang, Y., Xian, H., & Gao, L. (2015). Effects of driver behavior style differences and individual differences on driver sleepiness detection. *Advances in Mechanical Engineering, 7*(4). 1687814015578354.

Li, M., Zhang, C., & Yang, J.-F. (2010). An eeg-based method for detecting drowsy driving state. In Proceedings of seventh international conference on fuzzy systems and knowledge discovery*: 5* (pp. 2164–2167).

Lin, C.-T., Chang, C.-J., Lin, B.-S., Hung, S.-H., Chao, C.-F., & Wang, I.-J. (2010). A real–time wireless brain–computer interface system for drowsiness detection. IEEE Transactions on Biomedical Circuits and Systems, *4*(4), 214–222.

Lin, C.-T., Wu, R.-C., Liang, S.-F., Chao, W.-H., Chen, Y.-J., & Jung, T.-P. (2005). Eeg-based drowsiness estimation for safety driving using independent component analysis. IEEE Transactions on Circuits and Systems I: Regular Papers, *52*(12), 2726–2738.

Liu, H., Taniguchi, T., Tanaka, Y., Takenaka, K., & Bando, T. (2017). Visualization of driving behavior based on hidden feature extraction by using deep learning. IEEE Transactions on Intelligent Transportation Systems, *18*(9), 2477–2489.

McGehee, D. V., Mazzae, E. N., & Baldwin, G. S. (2000). Driver reaction time in crash avoidance research: Validation of a driving simulator study on a test track. In *Proceedings of the human factors and ergonomics society annual meeting: 44* (pp. 3–320). SAGE Publications Sage CA: Los Angeles, CA.

Meiring, G. A. M., & Myburgh, H. C. (2015). A review of intelligent driving style analysis systems and related artificial intelligence algorithms. *Sensors, 15*(12), 30653–30682.

Mitrovic, D. (2005). Reliable method for driving events recognition. IEEE Transactions on Intelligent Transportation Systems, *6*(2), 198–205.

Miyaji, M., Danno, M., & Oguri, K. (2008). Analysis of driver behavior based on traffic incidents for driver monitor systems. In Proceedings of the IEEE intelligent vehicles symposium (pp. 930–935).

Oliver, N., & Pentland, A. P. (2000). Graphical models for driver behavior recognition in a smartcar. In Proceedings of the IEEE intelligent vehicles symposium (pp. 7–12).

Rastgoo, M. N., Nakisa, B., Maire, F., Rakotonirainy, A., & Chandran, V. (2019). Automatic driver stress level classification using multimodal deep learning. *Expert Systems with Applications*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., … Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. International Journal of Computer Vision, *115*(3), 211–252.

Sabet, M., Zoroofi, R. A., Sadeghniiat-Haghighi, K., & Sabbaghian, M. (2012). A new system for driver drowsiness and distraction detection. In Proceedings of 20th Iranian IEEE conference on electrical engineering (pp. 1247–1251).

Sabokrou, M., Khalooei, M., Fathy, M., & Adeli, E. (2018). Adversarially learned one–class classifier for novelty detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3379–3388).

Sabokrou, M., Pourreza, M., Fayyaz, M., Entezari, R., Fathy, M., Gall, J., et al. (2018). Avid: adversarial visual irregularity detection. In *Asian conference on computer vision* (pp. 488–505). Springer.

Saito, Y., Itoh, M., & Inagaki, T. (2016). Driver assistance system with a dual control scheme: effectiveness of identifying driver drowsiness and preventing lane departure accidents. IEEE Transactions on Human-Machine Systems, *46*(5), 660–671.

Saradadevi, M., & Bajaj, P. (2008). Driver fatigue detection using mouth and yawning analysis. Internationsl Journal of Computer science and network security, *8*(6), 183–188.

Sathyanarayana, A., Boyraz, P., & Hansen, J. H. (2008). Driver behavior analysis and route recognition by hidden Markov models. In *Proceedings of IEEE international conference on vehicles electronics and safety* (pp. 276–281).

Sathyanarayana, A., Sadjadi, S. O., & Hansen, J. H. (2012). Leveraging sensor information from portable devices towards automatic driving maneuver recognition. In Proceedings of 15th internationsl IEEE conference on intelligent transportation systems (pp. 660–665).

Schnelle, S., Wang, J., Su, H.-J., & Jagacinski, R. (2016). A personalizable driver steering model capable of predicting driver behaviors in vehicle collision avoidance maneuvers. IEEE Transactions on Human-Machine Systems, *47*(5), 625–635.

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition* arXiv:1409.1556.

Škrjanc, I., Andonovski, G., Ledezma, A., Sipele, O., Iglesias, J. A., & Sanchis, A. (2018). Evolving cloud-based system for the recognition of drivers actions. *Expert Systems with Applications, 99*, 231–238.

Spiegel, S. (2015). Discovery of driving behavior patterns. In Smart information systems (pp. 315–343). Springer.

Spiegel, S., Jain, J.-B., & Albayrak, S. (2014). A recurrence plot-based distance measure. In *Translational recurrences* (pp. 1–15). Springer.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, *15*(1), 1929–1958.

Streiffer, C., Raghavendra, R., Benson, T., & Srivatsa, M. (2017). Darnet: A deep learning solution for distracted driving detection. In Proceedings of 18th ACM/IFIP/USENIX middleware conference: Industrial track (pp. 22–28). ACM.

Summala, H. (2000). Brake reaction times and driver behavior analysis. *Transportation Human Factors, 2*(3), 217–226.

Sun, C., Li, J. H., Song, Y., & Jin, L. (2014). Real-time driver fatigue detection based on eye state recognition. In *Applied mechanics and materials: 457* (pp. 944–952). Trans Tech Publ.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., … Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1–9).

Takeda, K., Hansen, J. H., Boyraz, P., Malta, L., Miyajima, C., & Abut, H. (2011). International large-scale vehicle corpora for research on driver behavior on the road. IEEE Transactions on Intelligent transportation systems, *12*(4), 1609–1623.

Tango, F., & Botta, M. (2013). Real-time detection system of driver distraction using machine learning. IEEE Transactions on intelligent transportation systems, *14*(2), 894–905.

UTDrive (2015). Utdrive. https://www.utdallas.edu/research/utdrive/index.html.

Van Ly, M., Martin, S., & Trivedi, M. M. (2013). Driver classification and driving style recognition using inertial sensors. In Proceedings of the IEEE intelligent vehicles symposium (iv) (pp. 1040–1045).

Wang, D., Pei, X., Li, L., & Yao, D. (2017). Risky driver recognition based on vehicle speed time series. IEEE Transactions on Human-Machine Systems, *48*(1), 63–71.
Wang, W., Xi, J., Chong, A., & Li, L. (2017). Driving style classification using a semisupervised support vector machine. IEEE Transactions on Human-Machine Systems, *47*(5), 650–660.
Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. arXiv:1505.00853.
Yan, S., Teng, Y., Smith, J. S., & Zhang, B. (2016). Driver behavior recognition based on deep convolutional neural networks. In *Proceedings of 12th international conference on natural computation, fuzzy systems and knowledge discovery* (pp. 636–641).
You, C., Lu, J., & Tsiotras, P. (2017). Nonlinear driver parameter estimation and driver steering behavior analysis for adas using field test data. IEEE Transactions on Human-Machine Systems, *47*(5), 686–699.
Yu, J., Chen, Z., Zhu, Y., Chen, Y. J., Kong, L., & Li, M. (2017). Fine-grained abnormal driving behaviors detection and identification with smartphones. IEEE Transactions on Mobile Computing, *16*(8), 2198–2212.
Zeiler, M. D. (2012). Adadelta: An adaptive learning rate method. arXiv:1212.5701.
Zeiler, M. D., & Fergus, R. (2013). *Visualizing and understanding convolutional networks* arXiv:1311.2901.
Zhu, B., Liu, Z., Zhao, J., Chen, Y., & Deng, W. (2018). Driver behavior characteristics identification strategies based on bionic intelligent algorithms. IEEE Transactions on Human-Machine Systems, (99), 1–10.