Abnormal Driving Detection with Normalized Driving Behavior Data: A Deep Learning Approach

Jie Hu, Xiaoqin Zhang, *Member, IEEE*, Stephen Maybank, *Fellow*, *IEEE*

***Abstract*—Abnormal driving may cause serious danger to both the driver and the public. Existing detectors of abnormal driving behavior are mainly based on shallow models, which require large quantities of labeled data. The acquisition and labelling of abnormal driving data are, however, difficult, labor-intensive and time-consuming. This situation inspires us to rethink the abnormal driving detection problem and to apply deep architecture models. In this study, we establish a novel deep-learning-based model for abnormal driving detection. A stacked sparse autoencoders model is used to learn generic driving behavior features. The model is trained in a greedy layer-wise fashion. As far as the authors know, this is the first time that a deep learning approach is applied using autoencoders as building blocks to represent driving features for abnormal driving detection. In addition, a method for denoising is added to the algorithm to increase the robustness of feature expression. The dropout technology is introduced into the entire training process to avoid overfitting. Experiments carried out on our self-created driving behavior dataset demonstrate that the proposed scheme achieves a superior performance for abnormal driving detection compared to the state-of-the-art.**

***Index Terms*—Abnormal driving detection, deep learning, stacked autoencoder.**

I. Introduction

TRAFFIC safety remains one of the main areas of study in vehicular technology. According to data released by World Health Organization, approximately 1.35 million people died in car crashes in 2018 [1]. In addition, more than 30 million people suffered non-fatal injuries, with many being disabled. Traffic accidents also caused considerable economic losses to individuals, their families, and nations. In many nations the economic loss is as much as 3% of their gross domestic product [1]. Studies show that the human factors, e.g., abnormal driving [2], account for around 95% of traffic accidents. Most drivers, however, fail to notice when they drive abnormally. It is therefore necessary to detect abnormal driving automatically and alert the driver for safe driving.

According to [3], abnormal driving is mostly caused by drunkenness, recklessness, and/or fatigue. With the rising popularity of smart phones, mobile use is now the leading cause of death behind the wheel [4]. Normal driving is characterized by good speed control of the vehicle and the avoidance of sudden acceleration [5]. Abnormal driving is characterized as follows: Firstly, drunk driving refers to driving while intoxicated by alcohol and is characterized by sudden acceleration, lack of control of the speed and long response time of 1.8~2.3*s* [6]. Secondly, it is shown in [3] that driving while fatigued is similar to drunk driving but without alcohol intoxication. Thirdly, unlike drunkenness and fatigue, the reckless driver is awake but might be impaired by mental factors. The driver behaves abnormally, with sudden accelerations and speeding [7]. Finally, drivers engaged in phone conversation slow down, and show an increase of 0.33*s* to 0.75*s* in the response time to external stimuli [8].

Abnormal driving detection (ADD) depends heavily on both real-time and historical driving behavior data collected by various sensors, i.e., throttle angle sensor, pressure sensor for the brake pedal, and vehicle-borne radar [9]. Due to the widespread use of traditional driving-related sensors and new emerging technologies of signal processing, unlabeled driving-related data are ubiquitous, and the transportation field has entered the era of big data. The management and modeling of driving behavior have thus become more data-driven. Although, many algorithms have been proposed for ADD, e.g., a back-propagation neural network (BPNN) [11], support vector machine (SVM) [12], [13], and dynamic Bayesian network [14]–[16], most of them use shallow driving models and require large quantities of labeled data. There is an aphorism in machine learning that states "sometimes it is not who has the best algorithm that wins; it is who has the most data"; one can always try to obtain more labeled data. The ground-truth labels of abnormal driving behavior, however, require expert knowledge to define and are very expensive to obtain. This inspires us to rethink the ADD problem and apply deep architecture models to take advantage of the large amount of unlabeled driving behavior data.

Recently, as a new state-of-the-art machine learning approach, deep learning [17] has received a great deal of attention in academe and industrial circles. It uses a multiple-layer architecture and is pretrained for extracting inherent features from huge amounts of unlabeled data. It is successfully applied in many domains, such as image restoration and recognition, dimension reduction, object detection, and human gaze estimation [10],[14],[18]-[20],[35]. As the driving process is complicated in real world, deep learning methods can represent driving features from unlabeled driving behavior data, where a good performance for ADD is obtained.

In this paper, we propose a novel deep-learning-based ADD system that can recognize four types of driving behaviors, i.e., normal, drunkenness/fatigue, recklessness, and phone use while driving, as shown in Fig. 1. First, in order to ensure that the different types of driving behavior are comparable, we normalize the abnormal driving behavior by employing a virtual driver to conduct the FTP-72 speed following task, which was described in our previous work [22]. Second, a stacked denoising sparse autoencoders (SdsAEs) model, which

J. Hu and X. Zhang are with the College of Computer Science and Artificial Intelligence, and also with the Institute of Big Data and Information Technology, Wenzhou University, Wenzhou 325035, China (e-mail: israel1987@126.com; zhangxiaoqinnan@gmail.com).

S. Maybank is with the School of Computer Science and Information Systems, Birkbeck College, London WC1E7HX, U.K. (e-mail: sjmaybank@dcs.bbk.ac.uk).

is trained in a layer-wise greedy fashion, is used to achieve unsupervised feature learning for ADD. The method of denoising can increase the robustness of feature expression, and thus improve the performance of the system. As far as the authors know, this is the first time that a deep learning approach is applied using autoencoders as building blocks to represent driving features for abnormal driving detection. A traditional classifier is then added on the top of the stack and the entire network is fine-tuned by an Error Propagation algorithm. In addition, the "dropout" technique is used during the training process to reduce the prediction error caused by overfitting. Finally, extensive experiments are undertaken to show the superior performance of the proposed scheme.

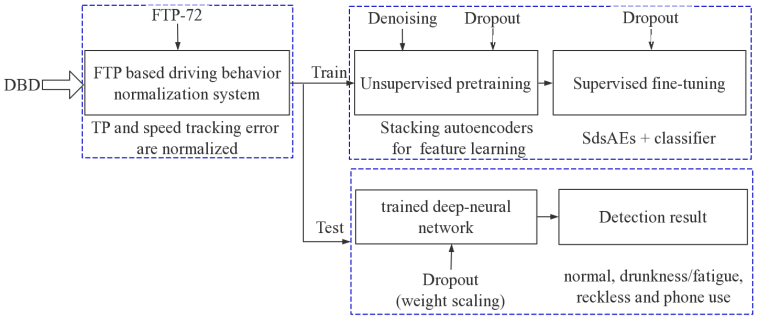


Fig. 1 Diagram of deep-learning-based method for ADD

The rest of this paper is organized as follows. Section II reviews the related work on the ADD. Section III introduces the general structure of driving behavior normalization. Section IV presents the deep-learning based ADD system. Section V discusses the experiments results. The conclusion is given in the last section.

II. Related Work

ADD has been accomplished using three different approaches: 1) driver monitoring systems, in which multiple electrodes are placed on a driver's skin to obtain physiological signals, such as electroencephalography, blood pressure, and blood alcohol level [23]; 2) computer vision based methods for measuring facial expressions, eye closure duration, yawning, or eyelid movement in real-time [24]; 3) driver-vehicle based information, such as throttle position (TP), brake pressure (BP), vehicle speed (VS), and steering [25]-[28]. Direct-contact-based methods (the first category) make the driver uncomfortable and are annoying, which might distract the driver. The methods in the second category are susceptible to external interferences, such as lighting changes and poor illumination at night, which can reduce accuracy. Methods in the third category are based on non-contact sensors and are robust to environmental factors. Thus, these methods are employed in this work. The most important related studies are summarized below.

Detection based on longitudinal driving behaviors, i.e., TP, BP, and VS. Yu *et al*. [25] propose a fine-grained abnormal driving behavior detection system based on 152 features related to the vehicle's acceleration and orientation; an SVM and BPNN were used for training. Sun *et al*. [26] use the gray relational analysis and proposed a novel fatigue driving detection method that incorporated 9 driving features in a two-level fusion structure. In [29], sensors such as an accelerometer, magnetometer, and Global Position System are employed and 15 features were extracted to detect high-risk motorcycle riding maneuvers.

Detection based on lateral driving behaviors, i.e., steering. By employing multivariate time series analysis, Li *et al*. [27] detect drunk driving based on lateral position and steering angle. To distinguish between sober and impaired drivers, Shirazi and Rad [28] employ system identification techniques, e.g., an average with exogenous input model to fit the data in order to describe the behavior of intoxicated drivers. The input and output of a driver are the lateral preview error and the steering wheel angle, respectively.

In summary, many algorithms for ADD have been developed on account of the growing demand for traffic safety and various techniques from different disciplines have been involved. However, it is difficult to say that one approach is significantly better than another in all cases. One reason for this is that the existing models for ADD are mainly developed based on a shallow network architecture, e.g., SVM, and the accuracy of ADD methods is dependent on the driving features embedded in the collected data.

With the continuous research on machine learning, neural networks have attracted more and more attention. The literature shows that deep-layered neural networks can learn more powerful models than shallow ones with a large amount of labeled data [19]. But the acquisition and labelling of abnormal driving data are difficult, labor-intensive and time-consuming. Moreover, the deep networks are hard to train with a straight forward Error Propagation algorithm. Recent advances in deep learning proposed by Hinton *et al*. [30] address these issues. The deep neural network can be pretrained by stacking autoencoders using a large amount of unlabeled data for better feature learning. The resulting networks have superior performance to the state-of-the-art. In this paper, we explore a deep-learning-based approach with SdsAEs for ADD.

III. Driving Behavior Normalization

The driving scenario is constituted by the driver, vehicle, and environment. Even a normal driver will occasionally behave abnormally, i.e., by frequent use of the gas pedal or steering wheel in a traffic jam. In order to detect abnormal driving

behavior more reliably, the environment should be standardized. Therefore, the driving behavior needs to be normalized in advance. In our previous work [22], we proposed a driving behavior normalization system based on driver model for driving style evaluation. The system is used as the basis for ADD in this work.

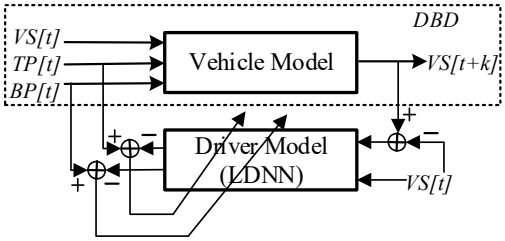


Fig. 2. Driver modelling based on direct inverse modelling and LDNN

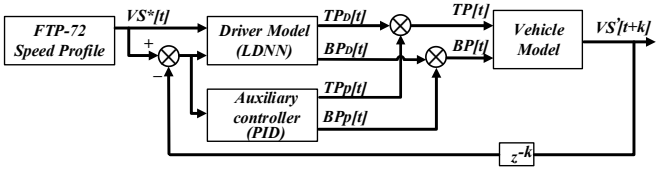


Fig. 3. Block diagram of driving behavior normalization

A. Driving Behavior Data

Driving behavior data (DBD) are collected by instrumented vehicles of the General Motors Corporation. The data take into account various combinations of traffic situations, vehicle types, and road conditions; nevertheless, they are unlabeled. Many different types of data are recorded, including VS, TP, engine speed, BP, and other data. Only VS, TP, and BP are used for ADD in this work for convenience. A total of 1036 unlabeled driving behavior samples are used with a time resolution of 0.1 s and a total distance of around 105 km for each sample.

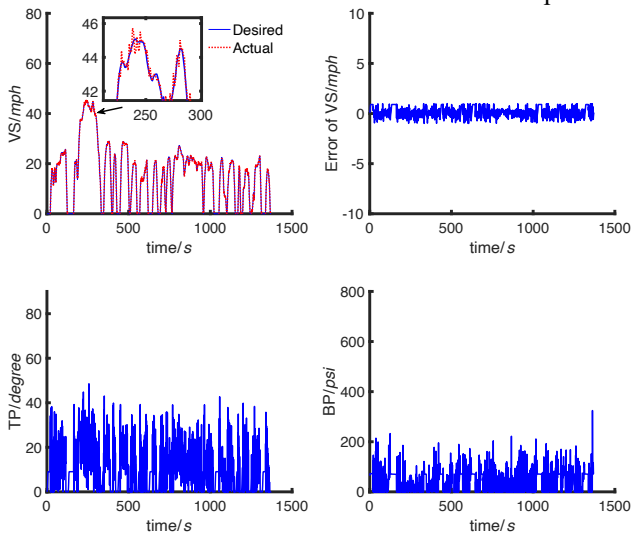


Fig. 4 An example of a speed-following result. Upper-left: FTP-72 speed profile (desired) versus following speed (model); upper-right: speed error; bottom-left: normalized TP; and bottom-right: normalized BP.

B. Driver Model

In the closed-loop control system formed by a driver and a vehicle, the driver tries to adapt to the vehicle gradually by learning to invert the vehicle model. Therefore, a driver model can be established based on a direct inverse modelling approach [29]. Due to their ability to tackle local variations in real-world data, e.g., DBD, the locally designed neural networks (LDNN), e.g., Cerebellar Model Articulation Controller (CMAC), are employed to make the modelling possible.

As shown in Fig. 2, adopted from a DBD sample, the VS, TP and BP at time t determine the VS at time $t+k$ due to a time lag. Therefore, the inputs of the driver model are $VS[t]$ and $VS[t+k] - VS[t]$, and the corresponding outputs are $TP[t]$ and $BP[t]$. The driver model is obtained by training the LDNN on the DBD.

C. Driving Behavior Normalization

As shown in Fig. 3, driving behavior normalization is implemented using a driver model which involves following the desired speed profile defined by FTP-72. The vehicle model is established by using a multilayer perceptron with a 3-7-1 structure for convenience [3]. As a common trick in learning control, a regular proportional-integral-derivative (PID) method is used as an auxiliary controller to accelerate the transient process. The gains of the PID are set as $\{K_p = 10, K_i = 0.5, K_d = 0.7\}$ in this work. At time t, due to the time lag k, $VS_{FTP}[t+k]$ is adopted from the FTP-72 speed profile to serve as the desired VS, or $VS^*[t]$. The final control signals, namely $TP[t]$ and $BP[t]$, are summations of the driver model and PID controller:

$$TP[t] = TP_D[t] + TP_P[t] \tag{1}$$

$$BP[t] = BP_D[t] + BP_P[t] \tag{2}$$

This normalization process assumes that every driver drives a vehicle in the same environment (FTP-72). As shown in Fig. 4, the speed-following task is successful accomplished and the driving behaviors, i.e., BP, TP, and VS are thus normalized for ADD.

IV. Methodology

Unsupervised pretraining using stacks of autoencoders followed by finetuning with an Error Propagation algorithm is an effective way of making use of unlabeled data. The well known deep learning model, SdsAEs, is introduced to extract the features that describe the driving behavior and construct a deep network for ADD [32].

A. Denoising sparse autoencoder

As an unsupervised three-layer neural network, an autoencoder [17] is formed by an encoder and decoder, as shown in Fig. 5. With an encoder, the input x is first mapped to a hidden representation h through a non-linearity f and then it mapped back into the original input x, as shown in

$$h = f(W^{(1)}x + b^{(1)}) \tag{3}$$

$$x = g(h) = f(W^{(2)}h + b^{(2)}) \tag{4}$$

where $\{W^{(1)}, b^{(1)}\}$ and $\{W^{(2)}, b^{(2)}\}$ are the weights of encoder and decoder, respectively; the sigmoid function $1/(1+\exp(-x))$ is considered for f and h in this paper. By minimizing the error $\|x - g(f(x))\|^2$ based on an Error Propagation algorithm, the optimal parameters are obtained. One issue concerned with autoencoders is that if the hidden layer has a larger size than the input, this approach could potentially learn an identity function.

Sparsity constraints can be added into the hidden layer to solve the problem.

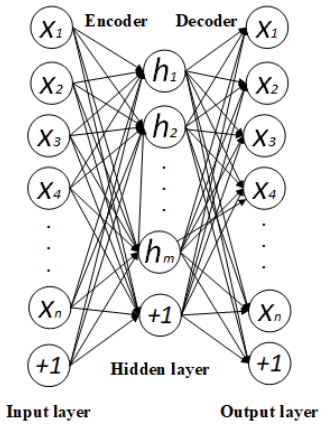


Fig. 5 Autoencoder

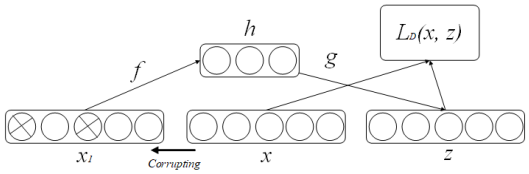


Fig. 6 The principle of denoising training

In addition, by adding noising to the input $\boldsymbol{X}$, a denoising sparse autoencoder is constructed for producing more stable and useful features and increasing the robustness of the model [33]. As shown in Fig. 6, an example x is stochastically corrupted to x_1, which is employed as the input of the autoencoder. The remaining part of the process is the same as an ordinary autoencoder. In this work, Gaussian noise was deemed appropriate due to the nature of the driving behavior data.

B. *Stacked denoising sparse autoencoders*

By stacking the denoising sparse autoencoders, the SdsAEs is established to form a deep network, where the output of the previous autoencoder is used as the input to the current layer. A greedy layer-wise based method proposed by Hinton *et. al*. [30] is employed for training SdsAEs.

In order to use the SdsAEs for ADD, a standard predictor is added on the top of layer. In this work, the softmax regression (SR) is put on the top of the stack for supervised abnormal driving behavior detection. The whole model is thus comprised by the SdsAEs and the predictor for ADD, as illustrated in Fig. 7.

C. *Dropout*

Overfitting is a serious problem in deep neural networks due to a large number of parameters and the limited training data. Dropout [34] is a technology for addressing this problem. As shown in Fig. 8, during the training process, each unit (only hidden layers in this work) is retained with a fixed probability p independent of other units. During testing, all units are always present and the weights are multiplied by p to make sure the output at test time is same as the expected output at training time.

D. *Training algorithm*

In this work, a SdsAEs model is constructed using the greedy layer-wise unsupervised training, where the network is initialized in a bottom-up way. After that, an Error Propagation algorithm is employed to fine tune the entire model in a top-down way to obtain better results.

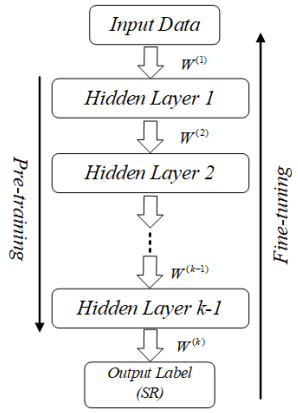


Fig. 7 Deep architecture model for abnormal driving detection

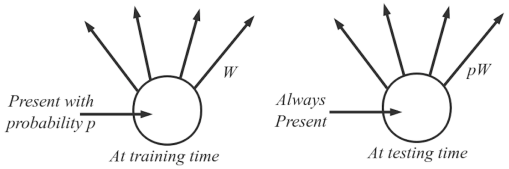


Fig. 8 Diagram of dropout

The training and testing of ADD are illustrated in Fig. 9. The procedure is summarized in Algorithm 1.

Algorithm 1. Training and testing ADD system

Step 1) Pretraining the *SdsAEs* model with unlabeled driving behavior data $\boldsymbol{x_1}$.

- Given the desired number of hidden layers l. Set the hyperparameters.
- Train the hidden layers based on greedy layer-wise method: take the training set as the input of the first hidden layer; for the nth hidden layer, the input is the output of the $(n\text{-}1)th$ hidden layer.
- Obtain $\{W^{(m)}, b^{(m)}\}$ by minimizing the objective function, where $0 \leqslant m \leqslant l$.

Step 2) Fine-tune the whole network with the labeled abnormal driving behavior data $\{\boldsymbol{x_2}, \boldsymbol{y_2}\}$

- Randomly initialize $\{W^{(l+1)}, b^{(l+1)}\}$.
- Use the Error Propagation algorithm to optimize the whole network's parameters in a top-down fashion.

Step 3) Test the system using testing data $\{\boldsymbol{x_3}, \boldsymbol{y_3}\}$

- Perform the forward propagation algorithm with input $\boldsymbol{x_3}$ to get the model output $\boldsymbol{y_3}^*$, where the weights are scaled as $W_{test} = pW$ as shown in Fig. 8.
- Calculate the detection accuracy by comparing the $\boldsymbol{y_3}^*$ and $\boldsymbol{y_3}$.

Note that the training and testing data are normalized by driving behavior normalization system in advance for reasonable ADD.

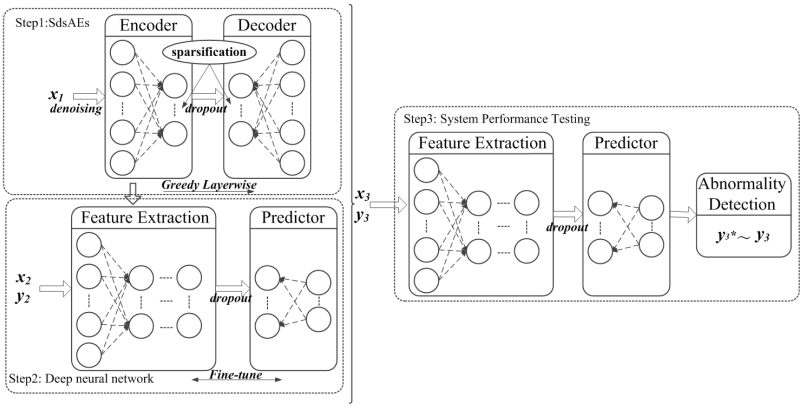


Fig. 9 Training and testing of the deep neural network

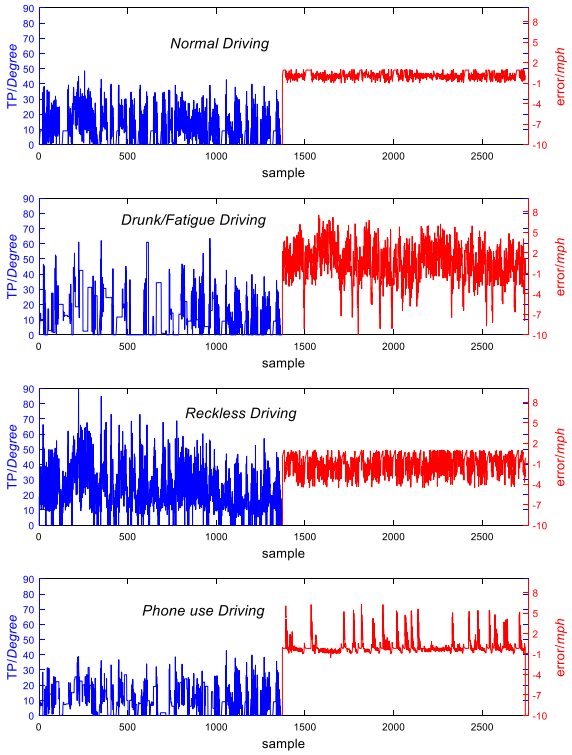


Fig. 10 Driving behavior patterns for four driving states.
(blue: normalized TP, red: normalized VS error)

V. EXPERIMENTS

A. Simulation of abnormal driving behavior

Since abnormal driving behavior data are difficult to collected and label, we simulate the data based on the collected DBD by considering the characteristics of TP and response delay as described in [4]. The description of the different types of abnormal driving behavior is presented in Section I and is listed briefly in Table I.

Table I
Driving behavior conditions and descriptions

Condition	Description
Normal	Good speed control, smooth acceleration
Drunk/fatigue	Poor speed control, rapid acceleration, response delay
Reckless	Speeding, rapid acceleration
Phone use	Lower speed, response delay

1) A drunk/fatigue driver usually exhibits rapid acceleration and deceleration with a response delay of 1.8 to 2.3 s, resulting in poor speed-following. We simulate this situation by delaying the TP for 2 * x seconds and raising its magnitude by (10 * x) % during FTP-72 testing.
2) A reckless driver usually exhibits sharp and rapid acceleration, even more so than a fatigued driver, resulting in frequently occurring excessive speed. We simulate this situation by raising the TP magnitude by (20 * x) %.
3) According to the result in [10], the drivers engaged in using a phone slow down and tend to delay frequent TP operation by average 0.7 s. We simulate this situation by delaying the TP for 0.7 * x seconds and decreasing its magnitude by (5 * x) %.

In order to simulate realistic abnormal driving behaviors, the value of x is sampled from a Beta distribution, i.e., $x \sim Be(5, 1)$.

In the distribution *Be*(5, 1), only the values in the range (0, 1) have non-zero probability density, and the probability density increases with the value.

Note that the driving behavior is contained in VS and TP [4]. To retain as much information as possible, the time series of the normalized TP and VS error are connected end to end and every point serves as an original input feature for ADD. Fig. 10 shows the simulated TP and VS error of a driver in normal and abnormal driving states, where the blue line represents the TP operation and the red represents the VS error. It is observed that diversities exist among the different types of driving behaviors. For the drunk/fatigued driver, the TP shows large fluctuations with considerable delays. The speed error is mostly in the range [-5, +5] *mph*. For the reckless driver, the TP changes considerably with a large magnitude. The speed error is in the range [-5, 0] *mph*. For the driver that uses the phone, the TP shows small fluctuations with a few delays. The speed error is in the range [0, +5] *mph*. The simulated abnormal driving behavior confirms the real-world driving.

In order to verify the effectiveness of the proposed deep-learning-based method for ADD, all DBD samples, in total 1036, are employed for abnormal driving behavior simulation. In total, 4144 samples are obtained, equally distributed among the four types of driving behavior (normal, drunk/fatigue, reckless, and phone use). 2072, 1036 and 1036 samples are randomly selected from the whole data set for training, validation and testing respectively. The training set is used for both pretraining and fine-tuning of the model and the validation set is used for choosing the best architecture. All the input data are normalized by driving behavior normalization and then scaled to a range (0, 1) according to Eq. (5), where f_{min} and f_{max} represent the maximum and minimum of the data.

$$f_s = \frac{f - f_{\min}}{f_{\max} - f_{\min}} \tag{5}$$

B. Determination of the hyperparameters

Table II
List of hyperparameters for deep network

hyperparameter	description	considered values
nHLayer	number of hidden layers	{1, 2, 3, 4}
nHUnit	number of units per hidden layer (same for all layers)	{100, 200, 300, 400, 500, 600, 700, 800, 900, 1000}
μ	noise level	standard deviation for Gaussian noise {0, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.45, 0.60, 0.75}
p	probability of retaining a unit for "dropout"	{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0}
lRate	learning rate for unsupervised pretraining and supervised fine-tuning	{0.05, 0.1, 0.15, 0.2}
nEpoq	number of training epochs for unsupervised pretraining and supervised fine-tuning	{10, 50, 100, 200, 350, 550, 800, 1100}

Some important architectural hyperparameters, including the number of input and output units, the number of hidden layers (*nHLayer*), the number of units per hidden layer (*nHUnit*), the noise level (μ), the probability of retaining a unit for dropout (*p*), learning rate (*lRate*), and number of training epochs (*nEpoq*) should be determined. We use the time series of TP and VS error as the input. Therefore, the dimension of the input space is 1372 * 2. For the output layer, the dimension is 4, including 1 normal driving behavior and 3 types of abnormal driving behavior. Table II lists the hyperparameters that should be tuned in this work based on validation set performance. Note that to reduce the choice space, the *nHUnit*, μ, and *p* are restricted to be the same for all hidden layers. *lRate* and *nEpoq* are also restricted to be the same for unsupervised pretraining and supervised fine-tuning. All hyperparameters are tuned for the best performance using grid search methods.

The experiments are conducted on a computer with 8-Core Intel Core i9 and PyCharm PROFESSIONAL 2017.3.7. The programming language Python and the TensorFlow library are used in the research.

1) *nHLayer* and *nHUnit*

We first examine how the proposed ADD strategy behaves as we increase the capacity of the deep NN both in breadth (*nHUnit*) and in depth (*nHlayer*) keeping other hyperparameters fixed (μ=0.15, *p*=0.4, *lRate*=0.15, *nEpoq*=800, ρ=0.05). Fig. 11 shows the evolution of the performance as *nHUnit* and *nHlayer* are increased. The validation error appears to come down until it reaches a critical point. The best performance is obtained when *hHLayer* is 3 and the *hHUnit* is around 500. The results confirm the lesson that the size of hidden layers of neural network should be neither too small nor too large [17].

2) corrupting noise level μ and dropout probability *p*

Fig. 12 shows the influence of the noise level μ. It can be seen that SdsAEs performs better than stacked sparse autoencoders (0 noise) for a rather wide range of noise levels (from 0.1 to 0.6). The best performance is obtained when the μ is close to 0.3. The model achieves the best performance under moderate denoising. A small denoising level is ineffective and a large level smooths away some important features.

The effect of changing dropout rates is also investigated. Note that the dropout is not applied in the input layer in this work. Fig. 13 shows the training and validation errors obtained as a function of *p*, the value *p* = 1 means dropout is absent. It can be seen that a small *p*, say 0.1, which ensures that very few units are turned on during training, leads to underfitting since the training error is high. The validation error decreases as *p* increases until *p* is close to 0.4. It then slowly increases and approaches a maximum as *p* becomes close to 1 (without dropout). The training error, on the other hand, decreases towards 0. These results confirm that moderate dropout is an effective technology to reduce the overfitting. And for the proposed scheme, the best probability of retaining a unit (*p*) is 0.4.

Furthermore, Fig. 14 shows the accuracies of three methods (with or without denoising and dropout) on different numbers of training samples. Two issues can be seen from Fig. 14.

- A comparison of the curves with the asterisk and circle indicates that the performance of the model is improved when the dropout is included during the training process. The performance improvement is more pronounced with a small amount of training data, e.g., 200. It confirms that the dropout technology is effective for reducing the risk of overfitting.

- A comparison of the curves with the block and asterisk indicates that adding noise to the input layer of the sparse autoencoder improves the ability to extract the intrinsic characteristics of abnormal driving behaviors, resulting in an improvement detection accuracy.

The effectiveness of adopting denoising and dropout method for training the deep network is thus further verified.

Other hyperparameters, i.e., *lRate*, *nEpoq*, are also tuned for best performance. The best structure of SdsAEs for ADD in this work is as shown in Table III.

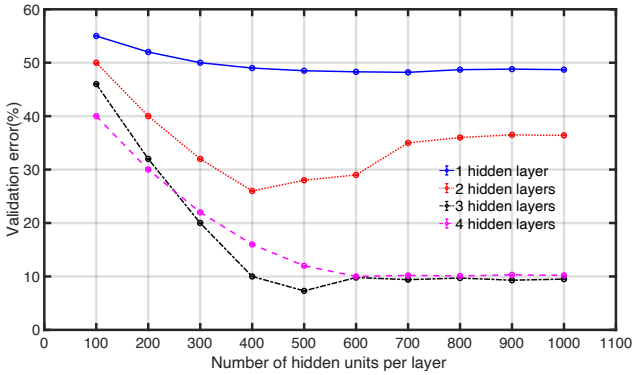


Fig. 11 Classification performance of ADD system on different *nHUnit* and *nHlayer*.

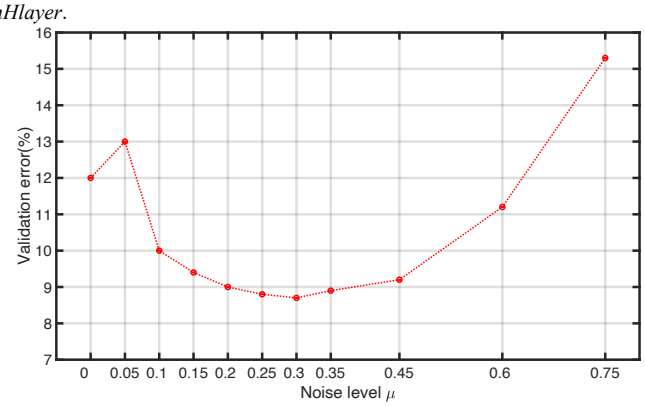


Fig. 12 Sensitivity to the noise.

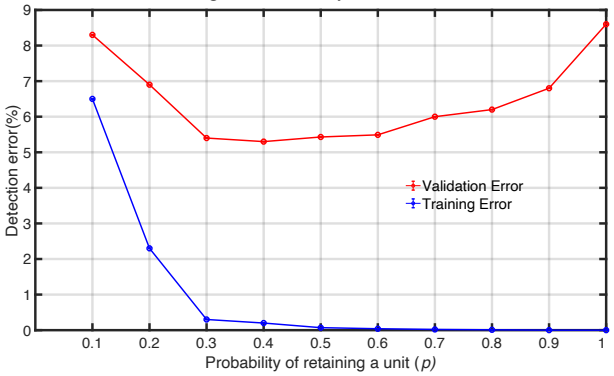


Fig. 13 Effect of changing dropout rates.

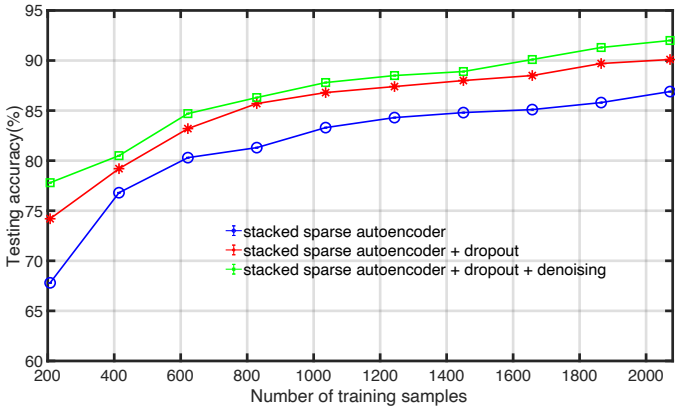


Fig. 14 Accuracies of three methods (with or without denoising and dropout).

Table III
Structure of SdsAEs for abnormal driving detection

nIUnit	*nHLayer*	*nHUnit*
2744	3	[500, 500, 500]
nOUnit	μ	p
4	0.3	0.4

C. *Detection results of the abnormal driving behavior*

Such indices as macro-average (macro-recall and macro-precision) are adopted to evaluate the performance of the proposed scheme on ADD. The definitions of these indices are as follows.

$$macro-recall = \frac{1}{4}\sum_{i=1}^{4}\frac{TP_i}{TP_i+FN_i} \quad (6)$$

$$macro-precision = \frac{1}{4}\sum_{i=1}^{4}\frac{TP_i}{TP_i+FP_i} \quad (7)$$

where TP_i is the number of class i driving behaviors correctly detected, FN_i is the number of class i driving behaviors misidentified and FP_i is the number of other classes of behaviors detected as class i.

The detection results of the four types of driving behavior based on the proposed scheme are shown in Table IV. Overall, the proposed scheme produces good results for ADD. The micro-recall and micro-precision are 92.66% and 92.69%, respectively. More specifically, for drunk/fatigued driving, 15 and 10 out of 259 samples are misclassified as reckless and phone use, respectively. For phone use driving, 14 samples are misclassified as drunk/fatigued driving. The reason for this might be that the drunk/fatigued driving is similar to phone use and reckless driving, i.e., longer reaction time to external stimuli as in phone use driving and rapid TP operation as in reckless driving. On the other hand, only 13 out of 777 abnormal samples are misclassified as normal (in bold), where the abnormal detection accuracy reaches up to 98.33%; hence, the proposed scheme is effective and promising for ADD.

Table IV
Abnormal driving detection performance of the proposed scheme

Actual \ Detected	Normal	Drunk/Fatigue	Reckless	Phone	Total	Index	
						Recall (%)	Precision (%)
Normal	249	1	7	2	259	96.14	95.04
Drunk/Fatigue	**4**	230	15	10	259	88.80	90.91
Reckless	**7**	8	243	1	259	93.82	90.00
Phone	**2**	14	5	238	259	91.89	94.82
Macro-average						92.66	92.69

Table V
Comprehensive Comparison on recall and (precision) %

Method	Driving behavior				Macro-average
	Normal	Drunk/ Fatigue	Reckless	Phone use	
SR	81.60 (80.04)	73.71 (74.21)	79.30 (77.82)	73.60 (74.87)	77.05 (76.74)
BPNN	85.19 (81.52)	72.20 (72.92)	82.37 (80.56)	75.42 (76.22)	78.80 (77.81)
SVM_{rbf}	94.24 (92.01)	**89.52** (87.30)	89.44 (86.10)	81.33 (84.23)	88.63 (87.41)
SdsAEs + Softmax layer	**96.14** (**95.04**)	88.80 (**90.91**)	**93.82** (**90.00**)	**91.89** (**94.82**)	**92.66** (**92.69**)

D. Comparison with other methods

Transverse comparison is conducted to verify the superior performance of the proposed scheme for ADD. We compared the performance of the SdsAEs model with the BPNN, SR, and SVM with RBF kernel (SVM_{rbf}). The BPNN is a common method for the ADD, as mentioned in Section II; the SR is the classifier layer adopted in this work; and the SVM_{rbf} is a relatively advanced algorithm for detection. Each model has been trained to the best architecture using grid search on the same data set as SdsAEs. The best architecture of BPNN consists of 2 hidden layers and 30 neurons per layer. For the SVM_{rbf} model, the C and γ are set to 2^{-1} and 2^{-11}, respectively.

The comparison results are given in Table V, where the best performance is in bold. The macro-recalls of SR, BPNN and SVM_{rbf} are 77.05%, 78.80% and 88.63% respectively compared to 92.66% in this study. The macro-precisions are 76.74%, 77.81% and 87.41% in the three ordinary methods and 92.69% in this study. Therefore, the SdsAEs systematically outperforms the other methods (except for SVM_{rbf} on the recall of drunk/fatigue driving, but the difference is not statistically significant). In addition, the performance gains of the SdsAEs compared to the SR of the same predictor are significant. They exhibit an average increase of 15% in the average. This shows that the proposed method that incorporates unsupervised pretraining can learn more robust features for ADD.

VI. Conclusion

A deep learning approach with SdsAEs for ADD has been investigated in this work. Unlike traditional methods with a shallow structure, the proposed deep network successfully detects different types of driving behavior. A greedy layer-wise unsupervised learning method is used to pre-train the deep network, followed by a fine-tuning process for updating the model parameters to improve the detection performance. By adding the noise, the autoencoders can learn more robust features so that the trained network has stronger anti-interference ability for ADD. A "dropout" technology is also used to reduce the risk of overfitting. The driving behaviors, i.e., TP, BP, and VS, are normalized by performing the FTP-72 driving test in advance to ensure appropriate results of the ADD. The best architecture of the proposed model is obtained by grid search method based on validation performance. The abnormal detection accuracy of this work reaches up to 98.33%, which verify the effectiveness and prospect of the proposed scheme for ADD. Transverse comparison demonstrates the superior performance of the proposed scheme.

The proposed scheme is effective and accurate for detecting abnormal driving and will be beneficial to the development of connected vehicle for safer and more comfortable driving. However, it is limited to longitudinal driving only, due to the limitation of the driving behavior data (DBD), which is essential to driving model. Other factors, such as weather, gender, etc., are not included in the DBD and hence are not considered in the driving modeling. In our future work, efforts will be made to detect abnormal driving behavior in more complicated conditions. Moreover, compressed sensing may be combined with the deep learning method [21, 36] employed in our work because there is strong correlation in the driving data.

VII. Acknowledgment

This work was supported in part by the National Natural Science Foundation of China [grant no. 61922064], in part by the Zhejiang Provincial Natural Science Foundation [grant nos. LR17F030001, LQ18F030010, LQ19F020005], in part by the Project of science and technology plans of Wenzhou City [grant nos. C20170008, G20150017, ZG2017016, G20170010].

References

[1]. World health organization, "Road traffic injuries," (2018). [Online]. Available: https://www.who.int/en/news-room/fact-sheets/detail/road-traffic-injuries.

[2]. J. Hu, *et al*., "Abnormal Driving Behavior Based on Normalized Driving Behavior," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 6645-6652, Aug. 2017.

[3]. S. Al-Sultan, A. H. Al-Bayatti and H. Zedan, "Context-aware driver behavior detection system in intelligent transportation systems," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 9, pp. 4264-4275, Nov. 2013.

[4]. L. Jin, *et al*., "Study on the impact degrees of several driving behaviors when driving while performing secondary task," *IEEE Access*, vol. 6, pp. 65772-65782, Oct. 2018.

[5]. M. Miyaji, M. Danno, and K. Oguri, "Analysis of driver behavior based on traffic incidents for driver monitor systems," in *Proc. IEEE Intell. Veh. Symp.*, Eindhoven, The Netherlands, Jun. 2008, pp. 930-935.

[6]. J. Rodriguez-Lopez, Y. Rebollo-Sanz, D. Mesa-Ruiz, "Hidden figures behind two-vehicle crashes: An assessment of the risk and external costs of drunk driving in Spain," *Accident Analysis and Prevention*, vol. 127, pp. 42-51, Jun. 2019.

[7]. T. Bar, *et al*., "Probabilistic driving style determination by means of a situation based analysis of the vehicle data," in *Proc. IEEE ITSC*, Washington, DC, USA, Oct. 2011, pp. 1698-1703.

[8]. M. Saifuzzaman, *et al*., "Impact of mobile phone use on car-following behavior of young drivers," *Accident Analysis and Prevention*, vol. 82, pp. 10-19, May, 2015.

[9]. W. Huang, *et al*., "Video-based abnormal driving behavior detection via deep learning fusions," *IEEE Access*, vol. 7, pp. 64571-64582, May, 2019.

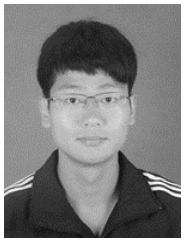
[10]. Xiaoqin Zhang, *et al*., "Attention-Based Interpolation Network for Video Deblurring", *Neurocomputing*, 2020.

[11]. Q. Liu, Z. Li, "Evaluation studies on ship driving fatigue based on BP artificial neural network," in *2012 8th International Conference on Natural Computation*, Chongqing, China, May, 2012, pp. 29-31.

[12]. J.F. Junior, et al., "Driver behavior profiling: an investigation with different smartphone sensors and machine learning," *Plos One*, vol. 12, no. 4, pp. 1-16, Apr. 2017.

[13]. S. Xie., *et. al*., "Nonstationary Linear Discriminant Analysis," in *Proceedings of the 51st Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Apr. 2018.

[14]. H. Shin, et al., "Stacked Autoencoders for Unsupervised Feature Learning and Multiple Organ Detection in a Pilot Study Using 4D Patient Data," *IEEE Transactions on Pattern Analysis and Machine Intelligent*, vol. 35, no. 8, pp. 1930-1943, Aug. 2013.
[15]. M. Imanl, S. F. Ghorelshi and U. M. Braga-Neto, "Bayesian Control of Large MDPs with Unknown Dynamics in Data-Poor Environments," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, Dec. 2018, pp. 8146-8156.
[16]. M. Imanl, *et. al*., "MFBO-SSM: Multi-Fidelity Bayesian Optimization for Fast Inference in State-Space Models," in *AAAI*, Jul. 2019.
[17]. Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn*., vol. 2, no. 1, pp. 1-127, 2009.
[18]. Y. Lv, *et al*., "Traffic Flow Prediction with Big Data: A Deep Learning Approach," *IEEE Transactions on Intelligent Transp*ortation Systems, vol. 16, no. 2, pp. 865-873, Apr. 2015.
[19]. A. Shrestha and A. Mahmood, "Review of deep learning algorithms and architectures," *IEEE Access*, vol. 7, pp. 53040-53065, May, 2019.
[20]. Y. Hu, M. Lu and X. Lu, "Driving behavior recognition from still images by using multi-stream fusion CNN," *Machine Vision and Applications*, vol. 30, no. 5, pp. 851-865, Jul. 2019.
[21]. Xiaoqin Zhang, *et al*., "Robust Low-Rank Tensor Recovery with Rectification and Alignment", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, July, 2020.
[22]. B. Shi, *et al*., "Evaluating Driving Styles by Normalizing Driving Behavior Based on Personalized Driver Modeling", *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 12, pp. 1502-1508, Dec. 2015.
[23]. M. Sakairi, "Water-cluster-detecting breath sensor and applications in cars for detecting drunk or drowsy driving," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1078-1083, May, 2012.
[24]. K. Murata, *et al*., "Noninvasive biological sensor system for detection of drunk driving," *IEEE Transactions on Information Technology in Biomedicine*, vol. 15, no. 1, pp. 19-25, Jan. 2011.
[25]. J. Yu, *et al*., "Fine-grained abnormal driving behaviors detection and identification with smartphones," *IEEE Transactions on Mobile Computer*, vol. 16, no. 8, pp. 2198-2212, Aug. 2017.
[26]. W. Sun, X. Zhang, and S. Peeta, "A real-time fatigue driving recognition method incorporating contextual features and two fusion levels," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 12, pp. 3408-3420, Dec. 2017.
[27]. Z. Li, X. Jin, X. Zhao, "Drunk driving detection based on classification of multivariate time series," *Journal of Safety Research*, vol. 54, pp. 61-67, Sep. 2015.
[28]. M. M. Shirazi, A. B. Rad, "Detection of intoxicated drivers using online system identification of steering behavior," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 4, pp. 1738-1747, Aug. 2014.
[29]. N. Condro, M. Li and R. Chang, "MotoSafe: active safe system for digital forensics of motorcycle rider with android," *International Journal of Information and Electronics Engineering*, vol. 2, no. 4, pp. 612-616, July, 2012.
[30]. G. H. Hinton, S. Osindero, and Y.-W. The, "A fast learning algorithm for deep belief nets," *Neural Computer*, vol. 8, no. 7, pp. 1527-1554, July, 2006.
[31]. L. Xu, *et al*., "Establishing Style-Oriented Driver Models by Imitating Human Driving Behaviors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2522-2530, Oct. 2015.
[32]. L. Meng, et. al., "Research of stacked denoising sparse autoencoder," *Neural Computer & Application*, vol. 30, pp. 2083-2100, Dec. 2018.
[33]. P. Vincent, et. al., "Extracting and composing robust features with denoising autoencoders," in *International Conference on Machine Learning*, Helsinki, Finland, July, 2008, pp. 1-16.
[34]. G. E. Hinton, et. al., "Improving neural networks by preventing co-adaptation of feature detectors," *Research Gate*, vol. 3, no 4, pp. 212-223, July, 2012.
[35]. Jiawei Xu, et. al., "A Bio-Inspired Motion Sensitive Model and Its Application to Estimating Human Gaze Positions under Classified Driving Conditions", *Neurocomputing*, 345(14): 23-35, 2019.
[36]. Yan Wu, et. Al, "Deep Compressed Sensing", *International Conference on Machine Learning*, pp. 6850-6860, 2019.



Jie Hu received the B.S and M.S. degrees in the Institute of Mechanical and Engineering from China Jiliang University, Hangzhou, China, in 2009, and 2012 respectively, and the ph.D. degree in electronic engineering from Zhejiang University, Hangzhou, China.

He is currently a lecturer with the College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou, China. His research interests include intelligent control and intelligent systems and industrial automation.

Xiaoqin Zhang received the B.Sc. degree in electronic information science and technology from Central South University, Changsha, China, in 2005, and the Ph.D. degree in pattern recognition and intelligent systems from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2010.

He is currently a Professor with the College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou, China. He has authored or coauthored more than 100 papers in international and national journals and international conferences. His research interests include pattern recognition, computer vision, and machine learning.

Steve Maybank received the B.A. degree in mathematics from King's College Cambridge, University of Cambridge, Cambridge, U.K., in 1976, and the Ph.D. degree in computer science from Birkbeck College, University of London, London, U.K., in 1988.

He is currently a Professor with the School of Computer Science and Information Systems, Birkbeck College, London, U.K. His current research interests are in the geometry of multiple images, camera calibration, visual surveillance, etc. Prof. Maybank is a Fellow of the Royal Statistical Society.