Materials Today: Proceedings 58 (2022) 251–255

Contents lists available at ScienceDirect

Materials Today: Proceedings

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

Driver distraction detection using machine learning techniques

Deepthi M. Pisharody [a,*], Binu P. Chacko [b], K.P. Mohamed Basheer [a]

[a] Department of Computer Science Sullamussalam Science College, Areekode 6736939, India
[b] Department of Computer Science Prajyoti Niketan College, Pudukad 680301, India

ARTICLE INFO

Article history:
Available online 16 February 2022

Keywords:
Driver distraction
SVM
Random forest
Naive Bayes
HOG

ABSTRACT

Number of deaths due to road accidents are increasing day by day. Although there are many reasons that may lead to road accidents, one primary reason is careless driving patterns followed by people on the road. These careless driving patterns need to be detected and avoided. In this article various machine learning techniques like Support Vector Machine (SVM), Random Forest, Naive Bayes method and an ensemble of above three are used to classify distracted driving patterns. SVM method gave better results than the other three. Green channel of the image is extracted and Histogram of Oriented Gradients (HOG) features are extracted. These features are applied to all machine learning methods in this paper. For the dataset used in this paper SVM method produced better results.

Selection and peer-review under responsibility of the scientific committee of the International Conference on Artificial Intelligence & Energy Systems.

1. Introduction

The major cause for death among youngsters is due to road accidents [1]. There may be many reasons for road accidents. Violation of traffic rules, distraction during driving or drowsiness while driving are major causes of all road accidents. Among these the most major cause is careless or distracted driving patterns followed by people while driving. This careless or distracted driving patterns is caused by involvement of drivers in secondary activities such as talking through mobile phones, texting on mobile phone, eating food, taking selfie, turning back and talking to co-passengers, doing hair or face makeup, using dashboard or stereo adjustment during driving, etc. Studies shows these secondary activities during driving are the riskiest reason for road accidents [2].

As per the Studies of National Highway Traffic Safety Administration (NHTSA), the usage of mobile phone while driving is one of the most common distracting activities of drivers [3].

In this paper three machine learning algorithms and an ensemble of three are applied to an image dataset containing various driving patterns created by Hesham et al. [4,5] and a comparative study is being conducted. Histogram of Oriented Gradients (HOG) features are extracted from green channel of images and applied to Machine Learning (ML) models. This dataset under study consists of 10,000 images under ten classes. Those classes are safe driving, talking through mobile phones by holding phone in right hand, talking through mobile phones by holding phone in left hand, texting on mobile phone using left hand, texting on mobile phone using right hand, turning back and talking to copassengers, using dashboard or stereo adjustment during driving, doing hair or face makeup, talking to co-passengers and drinking water.

This paper is organized into following sections. Section 2 provides literature review, Section 3 explains preliminary theories used in this paper, Section 4 compares study of different machine learning algorithms, Section 5 presents results and discussions and Section 6 concludes this article.

2. Literature review

Yuan Liao et al. [6] developed a method for detecting driver's cognitive dis- traction by using the Support Vector Machine Recursive Feature Elimination (SVM-RFE). SVM-RFE algorithm is used to extract feature subset and classification. The results of their method are measured in correction rate. The result in this article is measured as Correction rate and it is 95.8 4.4% for data collected at stop-controlled intersections and it is 93.7 5.0% for data collected in speed-limited highway.

Md Atiquzzaman et al. [7] detected two types of distracted driving pat- terns (texting distraction and eating/drinking distraction) using Linear Discriminant Analysis (LDA), Logistic Regression (LR), and Support Vector Machine. SVM could detect distractions with more accuracy than other two methods. SVM model detected tex-

* Corresponding author.
E-mail address: deepthi.saji@gmail.com (D.M. Pisharody).

https://doi.org/10.1016/j.matpr.2022.02.108
2214-7853/
Selection and peer-review under responsibility of the scientific committee of the International Conference on Artificial Intelligence & Energy Systems.

ting distraction with an accuracy of 84.33% and eating/drinking distraction with 79.53%. In addition to the above methods. They also proposed method using SVM, LDA, LR and RF (Random forests) algorithm also to detect eating/drinking distraction and texting distraction. RF algorithm could classify with better accuracy. Texting distraction is detected with an accuracy of 85.38% and that of eating distraction with an accuracy of 81.26%.

Tashrif Billah et al. [8] proposed a method for detecting various types of distractions while driving based on kernel support vector machine method. Relative distances between the tracking trajectories are used as features that represent actions of the driver. This method detected distracted driving pat terns with 91% accuracy and classify them into various groups with 90% accuracy. Renato Torres et al. [9] made a study to automatically classify drivers and passengers reading a message in a vehicle using models like SVM, DT (Decision Tree), LR, three ensemble learners - RF, ADM (AdaBoost Machine) and GBM (Gradient Boosting Machines), and one deep learning method Convolution Neural Network (CNN). The best performance was given by CNN and Gradient Boosting Machine models.

Tianchi Liu et al. [10] detected driver distraction using Semi-Supervised Extreme Learning Machine method. This method reduce cost since they used unlabelled data. Two driver states cognitively distracted or attentive is detected by using Laplacian support vector machine and semi-supervised extreme learning machine. The drawback of this method is that system detects distraction that uses data collected from the same driver in a particular driving scenario, since it uses unlabelled data.

Berri et al. [11] presented an SVM based model that detects the use of mobile phone while driving (i.e., distracted driving). The drawback of their method is that their segmentation algorithm failed when sunlight falls on driver. The average accuracy of their method is 91.57%.

Dina Kanaan et al. [12] built HMMs (Hidden Markov Models) from vehicle- based measures (in particular, GPS speed, steering wheel position, and lateral and longitudinal acceleration) to predict the existence of long off-path glances, secondary task engagement, and higher levels of motor control difficulty. Their model classified data into three predicted classes such as long off-path glances, secondary task engagement, and higher levels of motor control difficulty. GPS speed and steering wheel position were used to classify the two distraction indicators, whereas lateral and longitudinal acceleration were used to classify motor control difficulty. Their model could classify long- off path glances and secondary task engagement of drivers with accuracy of 77%.

Baheti et al. [13] proposed a CNN based approach to detect the distracted driver and to identify the cause of distraction. They used modified VGG-16 (Very Deep Convolutional Networks for Large-Scale Image Recognition) architecture that was proposed by Simonyan and Zisserman [14] in their paper. They could classify distraction with an accuracy of 95.54%. M Kumari et al. proposed a Convolutional Neural Network based algorithm to classify distracted faces of drivers [15]. Their model performed five categories of classification like no distraction, slight distraction to left, slight distraction to right, more distraction to left, more distraction to right. Their method worked with a classification accuracy of 93.631%. Eraqi et al. [4] proposed a model to detect and identify distraction that utilizes a genetically weighted ensemble of convolutional neural networks. Their model uses Multivariate Gaussian Naive Bayes classifier to classify skin and non-skin classes. To perform classification of hands and non-hand classes a trained model binary class AlexNet is used. Qunfang Xiong et al. [16] used PCN (Progressive Calibration Networks) for face detection, face tracking and tracking phone calling area in the face. Convolution neural network was used in their method for mobile phone detection. To identify call behavior of the driver they used modified YOLOV3 algorithm. They used deep learning approach to detect the use of mobile phones by drivers. The driver's face image is input to PCN algorithm to identify face area. Modified YoloV3 algorithm was used by them to detect the object (mobile phones). CNN was used to identify calling behavior of driver. The experimental results showed an accuracy of 96.56%, the false detection rate reaches 1.5%, and the processing speed reached 25 frames per second.

Whui kim et al. [17], in their method did labelling for face movement detection in 3 steps- initially face area is detected by using MTCNN (Multi-Task Cascaded Convolutional Neural Network) and, then crop the face and adjust its size. In the next step, choose reference cropped face to recognise face movements like left, right, top, bottom etc. They perform feature extraction using both Inception Resnet V2 network and Mobilenet. In both methods parameters are initialized using Xavier initialization and they did fine tuning in all layers of neural network. But the dataset created by them is purely indoor, there is no change in image properties like illumination, brightness, etc. Inception Resnet V2 network performed with a test accuracy of 92.1% and that the Mobilenet with 94.1%.

Hoang Ngan Le et al. [18] devised a Faster-RCNN (Region Based Convolutional Neural Networks) model to detect driver's cellphone usage and taking hands from the steering. Their model is mainly geared towards face/hand segmentation and classification. Shilpa Gite et al. [19] proposed eye gaze algorithm to track eye movements of drivers from video. They also proposed an improved recurrent neural network (RNN) model. An improved driver's movement tracking (DMT) algorithm was also proposed by them. This method track driver's movements and reduces noise. It shows 30% improvement time than normal machine learning algorithms. To recognise future action of driver they used RNN with LSTM (Long Short-Term Memory). But the drawback is that this method does just five classifications like left lane, right lane, turn left, turn right and safe driving, and all those are purely based on eye movements.

Andrei Aksjonov et al. [20] proposed a method for cellphone detection by using Nonlinear regression based on Euclidean distance (ED). To evaluate accuracy an adaptive neuro fuzzy inference system (ANFIS) is used in their method.

3. Proposed work

In this section, workflow of the proposed study is described. Fig. 1 is a sample dataset used for driver distraction detection created by H M Eraqi et al. which is used to conduct experiment in this paper. This dataset under study consists of 10,000 images under ten classes such as safe driving, talking through mobile phones by holding phone in right hand, talking through mobile phones by holding phone in left hand, texting on mobile phone using left hand, texting on mobile phone using right hand, turning back and talking to co-passengers, using dashboard or stereo adjustment during driving, doing hair or face makeup. talking to co-passengers and drinking water. Preprocessing, feature extraction, and learning and classification are the different stages in this work. Fig. 2 shows a workflow of this experiment and in the following subsections, each of these stages are explained elaborately.

Data obtained from dataset need to be preprocessed since it was of varying dimensions. In the first step of preprocessing all images were rescaled to a common dimension, 250×250 sized image. The

Fig. 1. Sample Dataset.

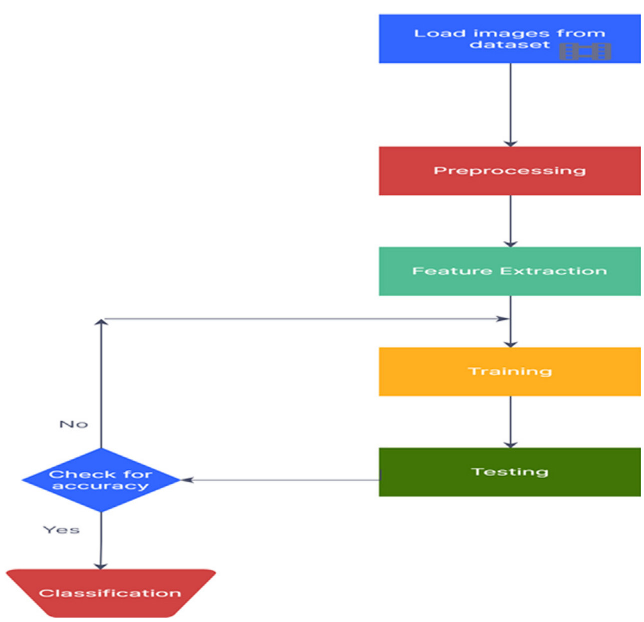


Fig. 2. Workflow of the method.

green channel of the colour image is obtained in the next step, since green channel gives more detailed information in an image.

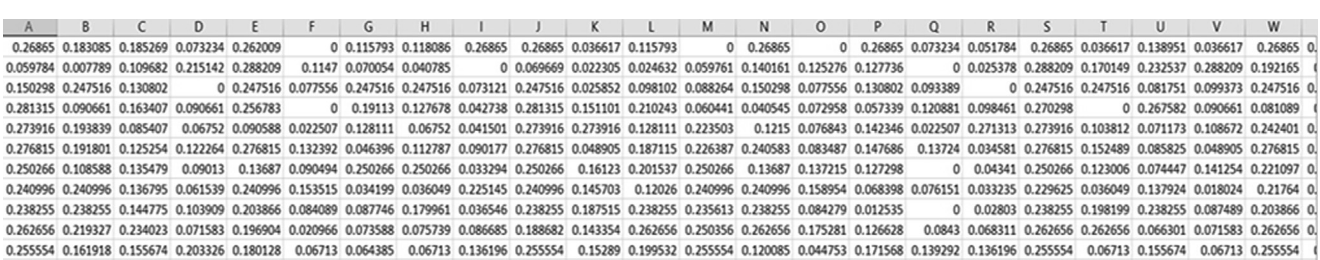
Histogram of Oriented Gradients, feature descriptor is used to extract features in this work. After feature extraction, the results are stored in a comma separated file (csv). Fig. 3 shows sample feature set obtained in this work.

The features extracted were fed into four classifiers like SVM, random forest, Naive Bayes classifier and ensemble of above three to identify the best classifier for this model.

4. Implementation and result analysis

The proposed work was implemented using the Python 3 programming language. The various libraries used for this experiment are sklearn for machine learning, OpenCV for computer vision and image processing, matplotlib for visualization, and other packages like NumPy, pandas etc. for scientific computations.

In the first step raw images from dataset were preprocessed and feature extraction was performed using HOG. The features extracted were fed into classifier models like SVM, random forest, Naive Bayes and an ensemble of three. The feature dataset was splitted into 2 in the ratio 7:3 (70% for training and 30% for testing). From the study conducted we found SVM is the best classifier for this work. The details of the classifier implementation and analysis are explained elaborately in the next section. Fig. 4 shows the confusion matrices of four models which is a summary of prediction



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
0.26865	0.183085	0.185269	0.073234	0.262009	0	0.115793	0.118086	0.26865	0.26865	0.036617	0.115793	0	0.26865	0	0.26865	0.073234	0.051784	0.26865	0.036617	0.138951	0.036617	0.26865
0.059784	0.007789	0.109682	0.215142	0.288209	0.1147	0.070054	0.040785	0	0.069669	0.022305	0.024632	0.059761	0.140161	0.125276	0.127736	0	0.025378	0.288209	0.170149	0.232537	0.288209	0.192165
0.150298	0.247516	0.130802	0	0.247516	0.077556	0.247516	0.247516	0.073121	0.247516	0.025852	0.098102	0.088264	0.150298	0.077556	0.130802	0.093389	0	0.247516	0.247516	0.081751	0.099373	0.247516
0.281315	0.090661	0.163407	0.090661	0.256783	0	0.19113	0.127678	0.042738	0.281315	0.151101	0.210243	0.060441	0.040545	0.072958	0.057339	0.120881	0.098461	0.270298	0	0.267582	0.090661	0.081089
0.273916	0.193839	0.085407	0.06752	0.090588	0.022507	0.128111	0.06752	0.041501	0.273916	0.273916	0.128111	0.223503	0.1215	0.076843	0.142346	0.022507	0.271313	0.273916	0.103812	0.071173	0.108672	0.242401
0.276815	0.191801	0.125254	0.122264	0.276815	0.132392	0.046396	0.112787	0.090177	0.276815	0.048905	0.187115	0.226387	0.240583	0.083487	0.147686	0.13724	0.034581	0.276815	0.152489	0.085825	0.048905	0.276815
0.250266	0.108588	0.135479	0.09013	0.13687	0.090494	0.250266	0.250266	0.033294	0.250266	0.16123	0.201537	0.250266	0.13687	0.137215	0.127298	0	0.04341	0.250266	0.123006	0.074447	0.141254	0.221097
0.240996	0.240996	0.136795	0.061539	0.240996	0.153515	0.034199	0.036049	0.225145	0.240996	0.145703	0.12026	0.240996	0.240996	0.158954	0.068398	0.076151	0.033235	0.229625	0.036049	0.137924	0.018024	0.21764
0.238255	0.238255	0.144775	0.103909	0.203866	0.084089	0.087746	0.179961	0.036546	0.238255	0.187515	0.238255	0.235613	0.238255	0.084279	0.012535	0	0.02803	0.238255	0.198199	0.238255	0.087489	0.203866
0.262656	0.219327	0.234023	0.071583	0.196904	0.020966	0.073588	0.075739	0.086685	0.188682	0.143354	0.262656	0.250356	0.262656	0.175281	0.126628	0.0843	0.068311	0.262656	0.262656	0.066301	0.071583	0.262656
0.255554	0.161918	0.155674	0.203326	0.180128	0.06713	0.064385	0.06713	0.136196	0.255554	0.15289	0.199532	0.255554	0.120085	0.044753	0.171568	0.139292	0.136196	0.255554	0.06713	0.155674	0.06713	0.255554

Fig. 3. Sample Feature Set.

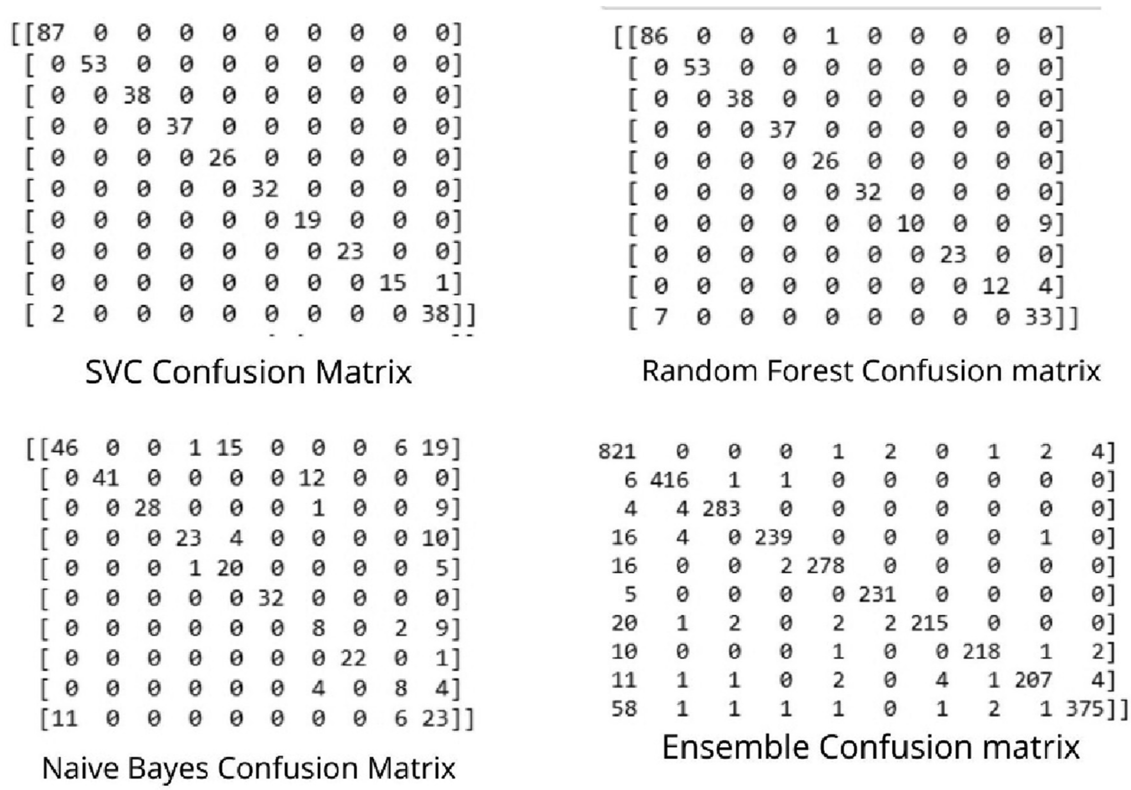


Fig. 4. Confusion matrices for various classifiers.

Table 1
Accuracy of various classifiers.

Model	SVM	Random Forest	Naive Bayes	Ensemble
Accuracy	99.19%	94.33%	67.65%	94.23%

results on a classification problem. The matrix summarizes the count of correct and incorrect predictions for each class. It is an important matrix to analyse the efficiency of machine learning algorithms.

Accuracy is one another parameter used for validating the classifier models. The accuracy of a classifier is a metric that measures the ratio of number of correct predictions to the total number of input samples. The accuracy obtained for the three models is shown in Table 1. From the table, it is clear that SVM classifier gives better accuracy of 99.19%. The accuracy obtained while using Naive Bayes classifier is very low compared to other methods. Naive Bayes classifier is a statistical classifier which works well if all the attributes are mutually independent. In the experiment conducted, it is impossible to state that all features are completely independent.

Another parameter for validating our model is F1 score. F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1]. It tells you how precise the classifier is as well as how robust it is. The greater the F1 Score, the better is the performance of our model. F1 score is calculated using formula: -

$$F1 = 2 \times ((precision \times recall)/(precision + recall)) \quad (7)$$

Precision is the number of True Positives divided by the number of True Positives and False Positives. Recall is the number of True Positives divide by the number of True Positives and the number of False Negatives. Support parameter can be defined as is number of actual occurrences of classes in the dataset under study. The value of support will not change based on the machine learning model. Fig. 5 plots classification summary of various models. This classification summary includes precision, recall, f1-score and support.

5. Conclusion

Accidents can be reduced if drivers are more concentrated on driving activities, avoiding secondary tasks while driving. Driver distractions need to be identified and warned the drivers. The proposed model was implemented as an application of Image Processing, Computer Vision, and Machine Learning to identify secondary tasks done by drivers while driving. Various machine learning models were implemented and studied to identify ten classes of driver distraction in this work. SVM classified data with higher accuracy of 99.19%. In this work only images were used to identify driver distraction. The classification works can be extended to deep

	precision	recall	f1-score	support
0	0.98	1.00	0.99	87
1	1.00	1.00	1.00	53
2	1.00	1.00	1.00	38
3	1.00	1.00	1.00	37
4	1.00	1.00	1.00	26
5	1.00	1.00	1.00	32
6	1.00	1.00	1.00	19
7	1.00	1.00	1.00	23
8	1.00	0.94	0.97	16
9	0.97	0.95	0.96	40

SVC Classification Report

	precision	recall	f1-score	support
0	0.92	0.99	0.96	87
1	1.00	1.00	1.00	53
2	1.00	1.00	1.00	38
3	1.00	1.00	1.00	37
4	0.96	1.00	0.98	26
5	1.00	1.00	1.00	32
6	1.00	0.53	0.69	19
7	1.00	1.00	1.00	23
8	1.00	0.75	0.86	16
9	0.72	0.82	0.77	40

Random Forest Classification Report

	precision	recall	f1-score	support
0	0.81	0.53	0.64	87
1	1.00	0.77	0.87	53
2	1.00	0.74	0.85	38
3	0.92	0.62	0.74	37
4	0.51	0.77	0.62	26
5	1.00	1.00	1.00	32
6	0.32	0.42	0.36	19
7	1.00	0.96	0.98	23
8	0.36	0.50	0.42	16
9	0.29	0.57	0.38	40

Naive Bayes Classification Report

	precision	recall	f1-score	support
0	0.85	0.99	0.91	831
1	0.97	0.98	0.98	424
2	0.98	0.97	0.98	291
3	0.98	0.92	0.95	260
4	0.98	0.94	0.96	296
5	0.98	0.98	0.98	236
6	0.98	0.89	0.93	242
7	0.98	0.94	0.96	232
8	0.98	0.90	0.93	231
9	0.97	0.85	0.91	441

Ensemble Classification Report

Fig. 5. Classification Report for various classifiers.

learning architectures like 3D CNN. In the next work video dataset can be used to identify and classify distractions. We are creating our own dataset to extend this work. Even techniques like transfer learning, few shot learning methods can be used to get better results with smaller datasets.

CRediT authorship contribution statement

Deepthi M. Pisharody: Conceptualization, Methodology ,Experimentation, Writing- Original Draft**Binu P. Chacko:** Reviewing and Editing, Supervsion. **K.P. Mohamed Basheer:** Writing- Reviewing and Editing, Supervsion.

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.

References

[1] "Global status report on road safety 2018 by WHO." https://www.who. int/publications/i/item/9789241565684.

[2] L. Barr, H. Howarth, S. Popkin, R. Carroll, J. Volpe, A review and evaluation of emerging driver fatigue detection measures and technologies". Proc. Int. Conf on fatigue management in transportation, 2005.

[3] NHSTA, "Driver electronic device use in 2010. traffic safety facts.." https://www.nhtsa.gov/sites/nhtsa.gov/, 2011.

[4] H.M. Eraqi, Y. Abouelnaga, M.H. Saad, M.N. Moustafa, Driver Distraction Identification with an Ensemble of Convolutional Neural Networks, J. Adv. Transport. 2019 (2019) 1–12.

[5] Y. Abouelnaga, H.M. Eraqi, M.N. Moustafa, Real-time distracted driver posture classification, arXiv preprint arXiv:1706.09498, 2017 [only preprint arxiv available].

[6] Y. Liao, S.E. Li, W. Wang, Y. Wang, G. Li, B. Cheng, Detection of driver cognitive distraction: A comparison study of stop-controlled intersection and speed-limited highway, IEEE Trans. Intell. Transport. Syst. 17 (6) (2016) 1628–1637.

[7] M.d. Atiquzzaman, Y. Qi, R. Fries, Real-time detection of drivers' texting and eating behavior based on vehicle dynamics, Transport. Res. Part F: Traffic Psychol. Behav. 58 (2018) 594–604.

[8] T. Billah, S.M. Rahman, M.O. Ahmad, M. Swamy, Recognizing distractions for assistive driving by tracking body parts, IEEE Trans. Circuits Syst. Video Technol. 29 (4) (2018) 1048–1062.

[9] R. Torres, O. Ohashi, G. Pessin, A machine-learning approach to distinguish passengers and drivers reading while driving, Sensors 19 (14) (2019) 3174.

[10] T. Liu, Y. Yang, G.-B. Huang, Y.K. Yeo, Z. Lin, Driver distraction detection using semi-supervised machine learning, IEEE Trans. Intell. Transp. Syst. 17 (4) (2015) 1108–1120.

[11] R.A. Berri, A.G. Silva, R.S. Parpinelli, E. Girardi, R. Arthur, A pattern recognition system for detecting use of mobile phones while driving vol. 2 (2014) 411–418.

[12] D. Kanaan, S. Ayas, B. Donmez, M. Risteska, J. Chakraborty, Using naturalistic vehicle-based data to predict distraction and environmental demand, Int. J. Mobile Human Computer Interaction (IJMHCI) 11 (3) (2019) 59–70.

[13] B. Baheti, S. Gajre, S. Talbar, Detection of distracted driver using convolutional neural network, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 1032–1038.

[14] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, 2014.

[15] M. Kumari, C. Hari, P. Sankaran, Driver distraction analysis using convolutional neural networks, in: 2018 Proc. Int. Conf on Data Science and Engineering (ICDSE), IEEE, 2018, pp. 1–5.

[16] Q. Xiong, J. Lin, W. Yue, S. Liu, Y. Liu, C. Ding, A deep learning approach to driver distraction detection of using mobile phone, in: 2019 IEEE Vehicle Power and Propulsion Conference (VPPC), IEEE Proc, 2019, pp. 1–5.

[17] W. Kim, H.-K. Choi, B.-T. Jang, J. Lim, Driver distraction detection using single convolutional neural network, in: 2017 Proc. Int. Conf on information and communication technology convergence (ICTC), IEEE, 2017, pp. 1203–1205.

[18] T. Hoang Ngan Le, Y. Zheng, C. Zhu, K. Luu, M. Savvides, Multiple scale faster-rcnn approach to driver's cell-phone usage and hands on steering wheel detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2016, pp. 46–53.

[19] S. Gite, H. Agrawal, K. Kotecha, Early anticipation of driver's maneuver in semiautonomous vehicles using deep learning, Progr. Artif. Intell. 8 (3) (2019) 293–305.

[20] A. Aksjonov, P. Nedoma, V. Vodovozov, E. Petlenkov, M. Herrmann, Detection and evaluation of driver distraction using machine learning and fuzzy logic, IEEE Trans. Intell. Transp. Syst. 20 (6) (2018) 2048–2059.