IJAMEC
e-ISSN: 2147-8228

INTERNATIONAL JOURNAL OF APPLIED MATHEMATICS ELECTRONICS AND COMPUTERS

www.dergipark.org.tr/ijamec

International Open Access

Volume 09
Issue 04

December, 2021

Research Article

Distracted Driving Detection with Machine Learning Methods by CNN Based Feature Extraction

Shafeeq Kanaan Shakir Al-Doori [a], ***Yavuz Selim Taspinar*** [b], ***Murat Koklu*** [c, *]

[a]*The Graduate School of Natural and Applied Science, Selcuk University, Konya, Turkey*
[b]*Doganhisar Vocational School, Selcuk University, Konya, Turkey*
[c]*Department of Computer Engineering, Selcuk University, Konya, Turkey*

ARTICLE INFO

Article history:
Received 12 December 2021
Accepted 22 December 2021

Keywords:
Classification
Deep learning
Driver distracted
Feature extraction
Machine learning

ABSTRACT

Millions of people lose their lives due to accidents caused by various reasons. As the number of vehicles increases, the number of accidents also increases. When driver errors caused by technological devices are added to this, the rate of accidents is increasing more and more. Generally, the vast majority of accidents occur as a result of distractions from drivers. For this reason, there is a need for a system based on the detection of driver errors and warning the driver in modern vehicles. For this purpose, the analysis of the convolutional neural network (CNN) feature extraction based classification models was carried out in this study. The SqueezeNet CNN architecture is trained with the transfer learning method and the image features are taken before the classification layer. The images were classified by giving the obtained features as input to k-nearest neighbour (k-NN), support vector machine (SVM) and random forest (RF) machine learning algorithms. 10-class dataset containing 22,424 driver error images was used in the training of the models. Classification successes of k-NN, SVM and RF models trained with images are 98.1%, 95.8%, and 88.7%, respectively. The highest classification success was obtained from the k-NN model. Other performance measurement metrics were also used for the detailed analysis of the classification models. It is aimed to find the most suitable model by comparing the training and testing times of the models. It is aimed that the obtained models can be used to detect driver errors over the image.

1. Introduction

Due to the high rate of accidents caused by driver errors, new integrations are made in modern vehicles in order to prevent accidents. Fatigue and distraction are among the biggest factors that cause drivers to make mistakes [1, 2]. Driving becomes a more challenging task with the increasing number of vehicles and population [3]. With the rapid development of technology, it is getting harder for drivers to focus on traffic. This situation increases driving performance and can lead to accidents. Examples of factors that keep drivers busy are talking to other passengers, mobile phones, navigation systems, and complex air conditioning systems [4].

Sagberg et al. revealed that insomnia causes mental disorders in drivers. They found that sleepless and tired accidents are more deadly than normal accidents. They explained that fatigue occurs as a result of physical and mental activities, and drowsiness is caused by insomnia [5]. Lal and Craig pointed out that blinking, yawning, mouth movements, head movements, and driving style are behaviors that can facilitate the detection of fatigue [6]. Bayly et al. suggested that accidents can be reduced by 20% by monitoring driver behavior. They stated that increasing road safety by detecting faulty behaviors and warning drivers is a possible solution that can reduce the number of accidents [7]. Dinges and Mallis stated that there are four types of numbness. These; These are mathematical, mission-ready technologies, vehicle performance, and in-vehicle driver-based approaches. In

[*] Corresponding author. E-mail address: *mkoklu@selcuk.edu.tr*
DOI: 10.18100/ijamec.1035749

the in-vehicle driver monitoring-based approach, computer vision and physiological signals are used. Thanks to the computer vision approach, the distraction of the drivers can be detected and warnings can be given to the driver [8]. Ranney et al stated in the literature that the symptoms of drowsiness are changes in the speed of blinking, decrease in concentration, change in driver's sitting position, steering stiffness, increase in yawning frequency, and changes in facial expressions. They stated that behavioral changes included confusion, delayed response, frequent breathing and frequent touching of the face [9]. Stutts et al. suggested that as in-vehicle monitoring technologies increase, so will focus-related accidents. They examined the contribution of different distractions [10].

Škrjanc et al. have proposed a cloud-based algorithm for detecting the behavior of drivers. They proposed a system based on the detection of maneuvers by monitoring the engine speed of the car, the speed of the car, the angle of the steering wheel. The experiments were carried out in a realistic simulator [11]. Wang et al. examined different hearing conditions of drivers. The data were obtained by virtual and real driving experiments. They created an emotion recognition model with fuzzy evaluation [12]. Olabiyi et al. proposed a system based on the idea of obtaining information about vehicle dynamics and driver with a camera in their proposed system. They classified the data they obtained with a bidirectional repetitive deep neural network. The proposed system can predict driver behavior before it performs [13]. Braunagel et al. used the data recorded while driving in the simulator environment of 73 people for automatic identification of driver behaviors. They stated that there are promising results in the recognition of in-vehicle driver behaviors with the proposed architecture [14]. Yan et al. proposed a deep learning-based method to detect driver behaviors. In the proposed method, images containing driver behaviors were used. They achieved 97.76% classification success with the proposed model. This method has proven to be an effective method for recognizing drivers' behavior [15]. Huang et al. proposed a hybrid convolutional neural network model to detect the behavior of distracted drivers. They suggested that the system they proposed increased driving activity [16]. Baheti et al. detected driver distraction by classifying the images. They proposed a new model called mobileVGG, based on deeply separable convolutions, with fewer parameters than other CNN models in the literature [17]. Mas et al. proposed a CNN+BiLSTM based model for detecting the spectral properties of images [18].

The aim of this study is to classify driver behaviors with convolutional neural networks using an image dataset [19] containing 10 different driver behaviors. In order to ensure the accuracy of the classification results, the features taken from the model trained using the SqueezeNet convolutional neural network were classified with 5 different machine learning methods. In this study, detailed analysis of the models was carried out in order to create models that can be used in practice.

The remainder of the article is organized as follows; dataset, machine learning methods, performance metrics are given in Chapter 2, experimental results are given in Chapter 3, and discussion and conclusions are given in Chapter 4.

2. Material and Methods

In this section, the dataset used in the study, machine learning methods used in classification, confusion matrix and performance evaluation are explained.

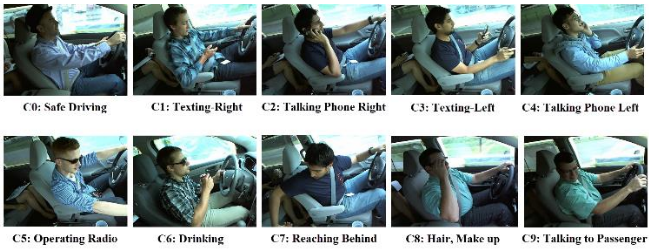
2.1. Dataset

An open access image dataset was used for training machine learning models by taking the features obtained as a result of training the SqueezeNet pre-trained model. The dataset used is Kaggle Challenge on Distracted Driver Detection by State Farm [19]. The data in the training folder of the dataset is labeled. The data in the test folder is not labeled. For this reason, images in the train folder containing 22,424 images were used in this study. There are 10 different classes of driver errors in the dataset. Classes are named from C0 to C9. The image properties included in the image classes in the dataset are shown in Table 1.

Table 1. The classes in the dataset, the types of images it contains, and the number of images by classes

Class	Distraction Type	Number of Images
C0	Safe Driving	2489
C1	Texting-Right	2267
C2	Talking Phone Right	2317
C3	Texting-Left	2346
C4	Talking Phone Left	2326
C5	Operating Radio	2312
C6	Drinking	2325
C7	Reaching Behind	2002
C8	Hair, Make up	1911
C9	Talking to Passenger	2129

Example images of each class in the dataset are shown in Figure 1.



C0: Safe Driving C1: Texting-Right C2: Talking Phone Right C3: Texting-Left C4: Talking Phone Left

C5: Operating Radio C6: Drinking C7: Reaching Behind C8: Hair, Make up C9: Talking to Passenger

Figure 1. Example images from the dataset by classes

2.2. Convolutional neural network (CNN)

They are architectures that are frequently used in image processing problems consisting of many layers. The images given as input to these architectures can be classified by extracting their properties [20]. The most important of the CNN layers is the convolution layer. In this layer, the features of the images are extracted with the help of filters. The features obtained after this layer are called activation maps. The convolution layer is followed by the nonlinear layer. Thanks to these layers, linear data is made non-linear, enabling neural networks to learn real-world problems. The pooling layer is used to reduce the number of nodes in the activation maps. It is usually placed between successive convolutional layers. Fully connected layers are used to provide connections between compute nodes. They usually form layers in the final stages of architectures [21]. They work like a kind of artificial neural network. Thanks to these layers, images are classified. In this study, SqueezeNet was used as the CNN architecture. This architecture has been preferred because its training and testing time is shorter than other architectures. A general representation of the CNN layers is given in Figure 2.

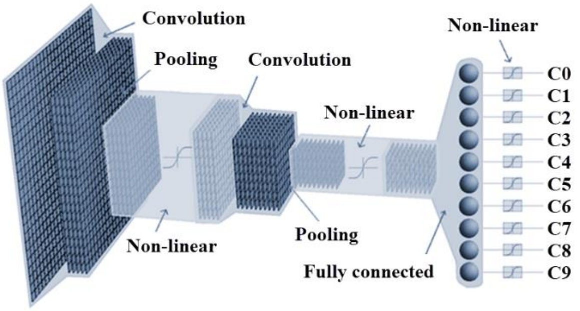


Figure 2. General representation of CNN layers

2.3. Transfer learning

The transfer learning approach is an approach that takes as an example people's use of their existing knowledge to solve new problems. With transfer learning, it is aimed to achieve higher success with less training data by using previous knowledge. Instead of training a model from scratch, models can be created to solve new problems with new training data by using previously trained models [22]. In this study, the pre-trained SqueezeNet model was used. Fine tuning was performed to train this model with new data. The training of the model was carried out by adapting the flatten10 layer, the layer before the last layer, which is the softmax layer, to the new class number. By using the weights of the pre-trained SqueezeNet model, the model in the study was created.

2.4. Machine learning methods

2.4.1. k-nearest neighbour (k-NN)

The k-nearest neighbour algorithm is a supervised machine learning algorithm that is frequently used in classification problems. In this method, the model is not trained using the data, instead it memorizes it. When a prediction is made, it performs the classification process by looking for the nearest neighbours of the newly incoming data. For this reason, it is used in classification problems as a fast classification algorithm. In the algorithm, a number k is determined. This k number indicates the number of neighbors. When a data to be estimated is received, the distance to the classes is calculated. After the distance is calculated, the classification process is carried out by assigning the data to the closest and most suitable class [23]. The number of k in the study was determined as 5.

2.4.2. Support vector machine (SVM)

It is a machine learning method that is frequently used in solving both classification and regression problem problems. SVM is used to solve two-class problems as well as multi-class problems. Multi-class SVM is called multi-SVM. In Multi-SVM, classification operations are performed by creating as many hyperplanes as the number of classes [24]. Kernel Radial Basis Function (RBF) was chosen in the SVM used in the study. The number of iterations was determined as 100.

2.4.3. Random forest (RF)

The RF algorithm is a supervised classification algorithm. In the algorithm, a random forest is first created. In the RF algorithm, finding the root node and splitting the nodes are done randomly. If there are enough trees in the algorithm, the probability of overfitting problem decreases. Trees for classification are constructed in such a way that each leaf node contains members of only one class. In this study, the RF algorithm was used for classification.

2.5. Confusion matrix and performance metrics

Confusion matrices are used to objectively evaluate the performance of models used in solving classification problems [26]. Confusion matrices have 4 different values obtained according to the classification results of the models. These are true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values [27]. A two-class confusion matrix is shown in Table 2.

Table 2. Two-class confusion matrix

		Predicted Class	
		P	**N**
Actual Class	**P**	TP	FN
	N	FP	TN

Evaluation of models is done with different metrics by using confusion matrix values. The metrics used in the study are Accuracy (ACC), F-1 score (FSC), Precision

(PSC), Recall (RCL) and Specificity (SPC). The calculation of these metrics is shown in Table 3.

Table 3. Calculation of performance metrics

Measure	Abbreviation	Formula
Accuracy	ACC	(TP+TN)/ (TP+TN+FP+FN)
Recall (r)	RCL	TP/(TP+FN)
Specificity	SPC	TN/(TN+FP)
Precision (p)	PSC	TP/(TP+FP)
F1-Score	FSC	(2*p*r)/(p+r)

The learning levels of the models can also be analyzed with the ROC curve and the AUC value, which is the area under this curve. The AUC value takes a value between 0-1. AUC value close to 1 indicates the level of learning of the model.

3. Experimental Results

Using the images in the dataset containing various driver error images, the features of the images were extracted with the SqueezeNet CNN model. The properties of the images are taken in the layer just before the softmax layer, which is the last layer of the SqueezeNet model. Each image has 1000 properties. The 1000 features obtained were given as input to 3 different machine learning algorithms. Figure 3 shows the flow chart showing the stages of the study.

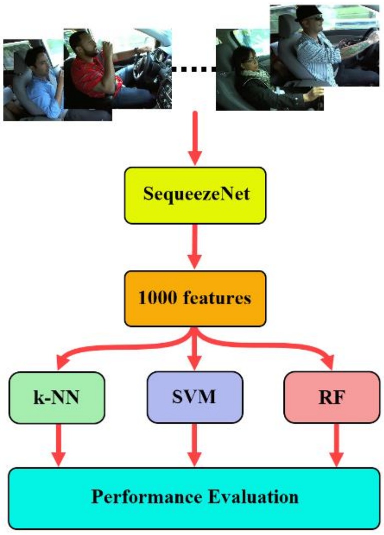


Figure 3. Flow chart of experimental process

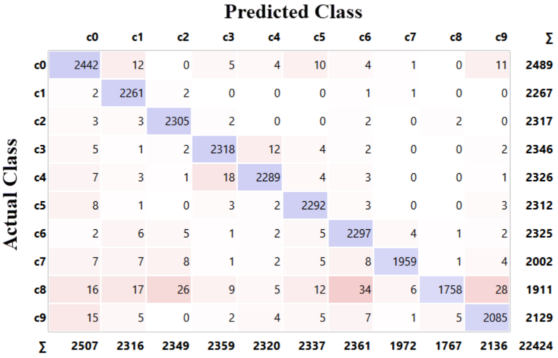
Training of k-NN, SVM and RF models was carried out. The training and testing times of the models are shown in Table 4.

Table 4. Training and testing time of all models

	Training Time (second)	Test Time (second)
k-NN	57.892	1372.018
SVM	3008.475	270.913
RF	64.061	2.212

According to Table 4, the fastest k-NN model was trained. However, the test time of the k-NN model has a high value. The lowest test time belongs to the RF model.

The dataset used in training the models is not divided into training-test. The cross validation method was used to objectively measure the performance of the models. In the cross validation method, the dataset is divided into k parts according to a predetermined k value. While k-1 piece is used in training, 1 piece is used in testing. This process is performed k times by changing the test pieces each time. The average classification success of the models is determined by taking the average of the classification success obtained in k numbers. In our study, the k value was determined as 10 and the models were evaluated. As a result of the training and testing of the models, a confusion matrix was obtained from each model. The confusion matrix obtained from the k-NN model is shown in Figure 4.

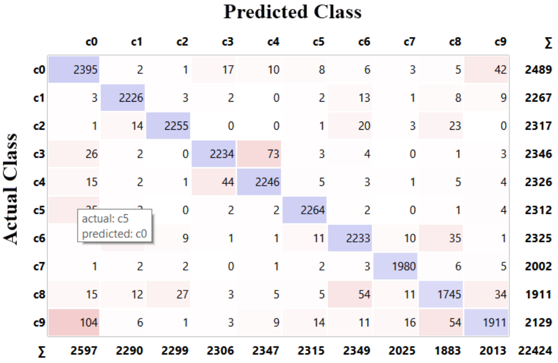


Predicted Class

Actual Class	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	∑
c0	2442	12	0	5	4	10	4	1	0	11	2489
c1	2	2261	2	0	0	0	1	1	0	0	2267
c2	3	3	2305	2	0	0	2	0	2	0	2317
c3	5	1	2	2318	12	4	2	0	0	2	2346
c4	7	3	1	18	2289	4	3	0	0	1	2326
c5	8	1	0	3	2	2292	3	0	0	3	2312
c6	2	6	5	1	2	5	2297	4	1	2	2325
c7	7	7	8	1	2	5	8	1959	1	4	2002
c8	16	17	26	9	5	12	34	6	1758	28	1911
c9	15	5	0	2	4	5	7	1	5	2085	2129
∑	2507	2316	2349	2359	2320	2337	2361	1972	1767	2136	22424

Figure 4. Confusion matrix of k-NN model

According to Figure 4, the most successfully classified class is C2: Talking Phone Right. The class classified with the lowest success is C8: Hair, Make up. Figure 5 shows the confusion matrix obtained from the SVM model.

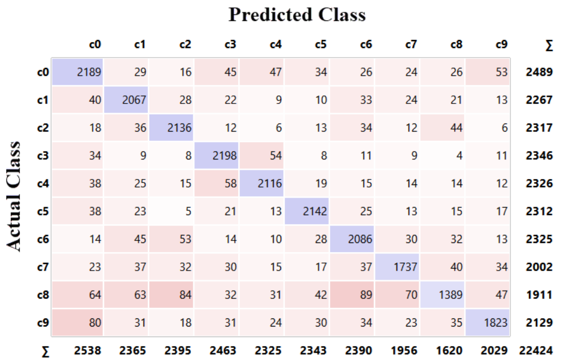


Predicted Class

Actual Class	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	∑
c0	2395	2	1	17	10	8	6	3	5	42	2489
c1	3	2226	3	2	0	2	13	1	8	9	2267
c2	1	14	2255	0	0	1	20	3	23	0	2317
c3	26	2	0	2234	73	3	4	0	1	3	2346
c4	15	2	1	44	2246	5	3	1	5	4	2326
c5	[illegible]	[illegible]	0	2	2	2264	2	0	1	4	2312
c6	[illegible]	[illegible]	9	1	1	11	2233	10	35	1	2325
c7	1	2	2	0	1	2	3	1980	6	5	2002
c8	15	12	27	3	5	5	54	11	1745	34	1911
c9	104	6	1	3	9	14	11	16	54	1911	2129
∑	2597	2290	2299	2306	2347	2315	2349	2025	1883	2013	22424

Figure 5. Confusion matrix of SVM model

According to Figure 5, the most successful class classified by the SVM model is C1: Texting-Right class. The class he classified with the lowest success is C9: Talking to Passenger. Figure 6 shows the confusion obtained from the RF model.


Predicted Class
Actual Class

	c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	Σ
c0	2189	29	16	45	47	34	26	24	26	53	2489
c1	40	2067	28	22	9	10	33	24	21	13	2267
c2	18	36	2136	12	6	13	34	12	44	6	2317
c3	34	9	8	2198	54	8	11	9	4	11	2346
c4	38	25	15	58	2116	19	15	14	14	12	2326
c5	38	23	5	21	13	2142	25	13	15	17	2312
c6	14	45	53	14	10	28	2086	30	32	13	2325
c7	23	37	32	30	15	17	37	1737	40	34	2002
c8	64	63	84	32	31	42	89	70	1389	47	1911
c9	80	31	18	31	24	30	34	23	35	1823	2129
Σ	2538	2365	2395	2463	2325	2343	2390	1956	1620	2029	22424

Figure 6. Confusion matrix of RF model

According to Figure 6, the RF model classified the C2: Talking Phone Right class most successfully. The lowest classification success belongs to C8: Hair, Make up class.

The performance metrics of the models were calculated using the data in the confusion matrices of all models. Performance metrics for all models are shown in Table 5.

Table 5. Performance metrics of all models

	ACC	FSC	PSC	RCL	SPC	AUC
k-NN	98.1	98.1	98.1	98.1	99.8	99.8
SVM	95.8	95.8	95.8	95.8	99.5	99.9
RF	88.7	88.6	88.6	88.7	98.7	98.4

According to Table 5, the highest classification success belongs to the k-NN model. Parallel to the classification success, the highest values in FSC, PSC, RCL and SPC metrics belong to the k-NN model. The highest AUC value belongs to the SVM model. ROC curves from all models are shown in Figure 7.

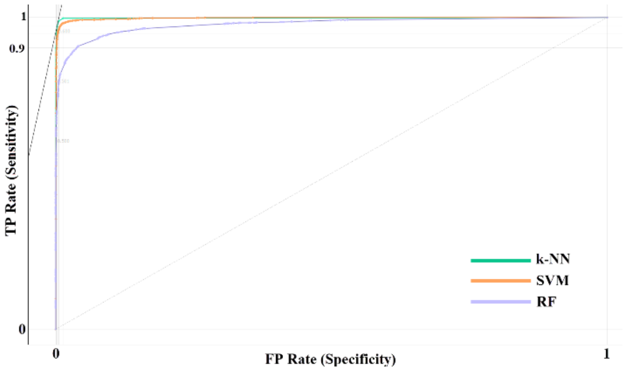


Figure 7. ROC curves of all models

4. Conclusions

In this study, images are classified using an image dataset containing driver errors. It is aimed to determine driver errors through images. For this purpose, a dataset containing a total of 22,424 images in 10 classes was used. The properties of the images were obtained from the SqueezeNet CNN model with the transfer learning method. Obtained features were classified by 3 different machine learning methods. k-NN, SVM and RF models were used in classification processes. As a result of the training of the models, a separate confusion matrix was obtained for each model. The performance metrics of the models were calculated using the data on the confusion matrices. The k-NN model has the highest classification success with 98.1%. The lowest classification success belongs to the RF model with 88.7%. The classification success of the SVM model was also determined as 95.8%. In other performance metrics, the highest values belong to the k-NN model. Training and testing times were also measured for detailed analysis of the classification models. The k-NN model was the fastest trained classification model with 57.892 seconds. The model with the longest training time is the SVM model with 3008.475 seconds. The model with the shortest test time is the RF model, with a test time of 2.212 seconds. The model with the longest test time is the k-NN model, with a test duration of 1372.018 seconds.

Considering the classification success of the models, the created models were very successful in detecting driver errors on the images. The proposed models will be used to detect real-time driver errors. In addition, it is thought that the classification success of machine learning models can be increased by increasing the images in the dataset. It is also possible to achieve higher classification successes with different machine learning models.

Acknowledgment

This paper was produced from Shafeeq Kanaan Shakir Al-Doori's master's thesis.

References

[1] Arnold, P. K., Hartley, L. R., Corry, A., Hochstadt, D., Penna, F., & Feyer, A. M., *Hours of work, and perceptions of fatigue among truck drivers.* Accident Analysis & Prevention, 1997. 29(4): p. 471-477.

[2] Philip, P., Sagaspe, P., Moore, N., Taillard, J., Charles, A., Guilleminault, C., & Bioulac, B., *Fatigue, sleep restriction and driving performance.* Accident Analysis & Prevention, 2005. 37(3): p. 473-478.

[3] Beirness, D.J., H.M. Simpson, and A. Pak, *The road safety monitor: Driver distraction*. 2002.

[4] Wahlstrom, E., O. Masoud, and N. Papanikolopoulos. *Vision-based methods for driver monitoring*. in *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*. 2003. IEEE.

[5] Sagberg, F., Jackson, P., Krüger, H. P., Muzet, A., & Williams, A. J., *Fatigue, sleepiness and reduced alertness as risk factors in driving*. 2004: Institute of Transport Economics Oslo.

[6] Lal, S.K. and A. Craig, *A critical review of the psychophysiology of driver fatigue.* Biological psychology, 2001. 55(3): p. 173-194.

[7] Bayly, M., Fildes, B., Regan, M., & Young, K., *Review of crash effectiveness of intelligent transport systems.* Emergency, 2007. 3: p. 14.

[8] Dinges, D. and M. Mallis. *Managing fatigue by drowsiness detection: Can technological promises be realized?* in

International Conference on Fatigue and Transportation, 3rd, 1998, Fremantle, Western Australia. 1998.

[9] Ranney, T.A., W.R. Garrott, and M.J. Goodman, *NHTSA driver distraction research: Past, present, and future*. 2001, Citeseer.

[10] Stutts, J. C., Reinfurt, D. W., Staplin, L., & Rodgman, E., *The role of driver distraction in traffic crashes.* 2001.

[11] Škrjanc, I., Andonovski, G., Ledezma, A., Sipele, O., Iglesias, J. A., & Sanchis, A., *Evolving cloud-based system for the recognition of drivers' actions.* Expert Systems with Applications, 2018. 99: p. 231-238.

[12] Wang, X., Liu, Y., Wang, F., Wang, J., Liu, L., & Wang, J., *Feature extraction and dynamic identification of drivers' emotions.* Transportation research part F: traffic psychology and behaviour, 2019. 62: p. 175-191.

[13] Olabiyi, O., Martinson, E., Chintalapudi, V., & Guo, R., *Driver action prediction using deep (bidirectional) recurrent neural network.* arXiv preprint arXiv:1706.02257, 2017.

[14] Braunagel, C., Kasneci, E., Stolzmann, W., & Rosenstiel, W., *Driver-activity recognition in the context of conditionally autonomous driving*. in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. 2015. IEEE.

[15] Yan, S., Teng, Y., Smith, J. S., & Zhang, B. *Driver behavior recognition based on deep convolutional neural networks*. in *2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*. 2016. IEEE.

[16] Huang, C., Wang, X., Cao, J., Wang, S., & Zhang, Y., *HCF: a hybrid CNN framework for behavior detection of distracted drivers.* IEEE Access, 2020. 8: p. 109335-109349.

[17] Baheti, B., S. Talbar, and S. Gajre, *Towards computationally efficient and realtime distracted driver detection with mobilevgg network.* IEEE Transactions on Intelligent Vehicles, 2020. 5(4): p. 565-574.

[18] Mase, J. M., Chapman, P., Figueredo, G. P., & Torres, M. T., *A hybrid deep learning approach for driver distraction detection*. in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*. 2020. IEEE.

[19] *State Farm. Distracted Driver Detection Competition.* [cited 2021 9 December];
Available from: https://www.kaggle.com/c/state-farm-distracted-driver-detection.

[20] Koklu, M., I. Cinar, and Y.S. Taspinar, *Classification of rice varieties with deep learning methods.* Computers and Electronics in Agriculture, 2021. 187: p. 106285.

[21] Koklu, M., I. Cinar, and Y.S. Taspinar, *CNN-based bi-directional and directional long-short term memory network for determination of face mask.* Biomedical Signal Processing and Control, 2022. 71: p. 103216.

[22] Taspinar, Y.S., I. Cinar, and M. Koklu, *Classification by a stacking model using CNN features for COVID-19 infection diagnosis.* Journal of X-Ray Science and Technology, 2021(Preprint): p. 1-16.

[23] Ali, M., Jung, L. T., Abdel-Aty, A. H., Abubakar, M. Y., Elhoseny, M., & Ali, I., *Semantic-k-NN algorithm: an enhanced version of traditional k-NN algorithm.* Expert Systems with Applications, 2020. 151: p. 113374.

[24] Yan, X. and M. Jia, *A novel optimized SVM classification algorithm with multi-domain feature and its application to fault diagnosis of rolling bearing.* Neurocomputing, 2018. 313: p. 47-64.

[25] Speiser, J. L., Miller, M. E., Tooze, J., & Ip, E., *A comparison of random forest variable selection methods for classification prediction modeling.* Expert systems with applications, 2019. 134: p. 93-101.

[26] Yurttakal, A. H., Erbay, H., İkizceli, T., Karacavus, S., & Çinarer, G., *A comparative study on segmentation and classification in breast mri imaging.* IIOAB journal, 2018. 9(5): p. 23-33.

[27] Yasar, A., E. Kaya, and I. Saritas, *Classification of Wheat Types by Artificial Neural Network.* International Journal of Intelligent Systems and Applications in Engineering, 2016. 4(1): p. 12-15.