



Machine Learning for Cognitive Load Classification – A Case Study on Contact-Free Approach

Mobyen Uddin Ahmed^(✉), Shahina Begum, Rikard Gestlöf, Hamidur Rahman,
and Johannes Sörman

School of Innovation Design and Engineering (IDT), Mälardalen University, Västerås, Sweden
{mobyen.ahmed, shahina.begum, hamidur.rahman}@mdh.se

Abstract. The most common ways of measuring Cognitive Load (CL) is using physiological sensor signals e.g., Electroencephalography (EEG), or Electrocardiogram (ECG). However, these signals are problematic in situations e.g., in dynamic moving environments where the user cannot relax with all the sensors attached to the body and it provides significant noises in the signals. This paper presents a case study using a contact-free approach for CL classification based on Heart Rate Variability (HRV) collected from ECG signal. Here, a contact-free approach i.e., a camera-based system is compared with a contact-based approach i.e., Shimmer GSR+ system in detecting CL. To classify CL, two different Machine Learning (ML) algorithms, mainly, Support Vector Machine (SVM) and k-Nearest-Neighbor (k-NN) have been applied. Based on the gathered Inter-Beat-Interval (IBI) values from both the systems, 13 different HRV features were extracted in a controlled study to determine three levels of CL i.e., S0: low CL, S1: normal CL and S2: high CL. To get the best classification accuracy with the ML algorithms, different optimizations such as kernel functions were chosen with different feature matrices both for binary and combined class classifications. According to the results, the highest average classification accuracy was achieved as 84% on the binary classification i.e. S0 vs S2 using k-NN. The highest F_1 score was achieved 88% using SVM for the combined class considering S0 vs (S1 and S2) for contact-free approach i.e. the camera system. Thus, all the ML algorithms achieved a higher classification accuracy while considering the contact-free approach than contact-based approach.

Keywords: Cognitive Load (CL) · Contact-free approach · k-Nearest-Neighbor (k-NN) · Support Vector Machines (SVM) · Machine Learning (ML)

1 Introduction

Driving is a complex task that requires high concentration with simultaneous skills and abilities [1]. Research shows that, most of the traffic accidents are caused by the human error. [1]. Most of the cases a road accident occurs when a driver loose concentration on

driving due to look outside or thinking anything rather than driving task. Therefore, there is a connection between low concentration level and traffic accidents has been identified in [2]. Research shows that the number of road deaths increased by 28% in Sweden, by 14% in the Czech Republic and by 11% in the Netherlands in 2018 compared to 2017 [3]. Over 20% of those 90% are due to fatigue or CL (e.g. use of in-vehicle devices which can lead to cognitive overload for the driver) [4]. CL is also known as cognitive workload, which is a concept that describes the relationship between the cognitive demands from a task and the environment that influence the user's cognitive resources [5, 6].

To identify CL, psycho-physiological measures are mainly used, as they assess interactions between physiological and psychological states of a human. The most common psychophysiological measurement to detect CL is EEG, because it objectively identifies the cognitive cost of performing tasks [6]. Other parameters are also used to detect CL, such as: pupil dilation, HRV, galvanic skin response (GSR), and EEG etc. [7]. There are disadvantages with these contact-based approaches as most of them used electrodes and they should be connected to the human body, which can be cumbersome to use in situations such as driving. In many occasions the measurement signals are affected due to movements and artifacts [8]. That is why a contact-free way of measuring CL could be ideal for these situations. On the contrary, the contact-free approach e.g. camera systems have been incessantly growing in the research community [9]. Here, the camera system have developed based on color schemas such as RGB and the Lab color space to extract heart rate (HR) and IBI by detecting variation in facial skin-color caused by cardiac pulse [9]. McDuff et al. [10] has built a remote person independent classifier and predict CL with 85% classification accuracy, here, a contact-free camera-based system was used in order to measure HR, HRV and breathing rate (BR).

In order to detect or classify CL, HRV features are used, since the HR fluctuates with varying levels of CLs which can be detected with HRV metrics [11]. In medical diagnoses and stress detection HRV is a commonly used parameter, there are many different features in HRV that can be calculated and can be used as a parameter to estimate CL. There are three established ways to calculate the HRV, either by time-domain, frequency-domain or non-linear measurements [12]. CL detection is a very inefficient for humans to manually review large data sets because of the factors that occur naturally in everyday life situations [13]. This is why the ML algorithms are an essential factor in classifying CL.

The data collection for cognitive load research are generally conducted either in indoor environment using simulator driving or in outdoor environment using real road driving scenario [14–16]. The authors in [17] have focused on the feasibility of using visual attention features to classify CL in three different states i.e., low, medium and high cognitive load. Four different classification models such as k-NN, logistic regression, SVM, adaptive boosting and random forest have used and the k-NN received the highest accuracy of 81%. A study performed by Wang et al. [18] has received a 97.2% of classification accuracy in detecting high CL.

This paper presents a case study on contact-free approach for CL classification based on the features extracted through HRV analysis and using on ML. Here, the goal of the study is to compare the contact-free approach i.e., camera-based system with the contact-based approach i.e., a Shimmer GSR+ system to determine how well the

contact-free approach performs in detecting CL. To classify the CL, two different ML algorithms, mainly, SVM and k-NN are used. This study includes a collection of data for both the systems in a controlled environment. Thirteen HRV features were extracted to determine three states of CL based on the gathered IBI values from both the systems. The three levels are, S0: reference point for low CL, S1: normal CL where the test subjects performed easy puzzles and drove normally in a video game and S2: high CL where the test subjects completed hard puzzles and drove on the hardest course of a video game while answering math questions [19]. To get the best classification accuracy with the ML algorithms, different optimizations such as kernel functions i.e., radial basis function (RBF), and sigmoid function were chosen with different feature matrixes both for binary and combined class classifications.

2 Materials and Methods

In the contact-free approach, a Logitech HD Webcam C615 camera with the resolution of 1920×1080 , and a frame rate of 30fps equipped with auto focus was used to record faces of the test subjects. The recorded videos were saved in MP4 format. The camera was adjusted according to the height of the head of the test subject so that it can capture the facial image properly. The test subjects were sat roughly 70 cm in front of the camera during the different phases of the controlled experiment, thus the desired ROI of the test subjects and facial features were extracted [9]. During the data collection, the test subjects were asked to move their heads as normal but not too much (Fig. 1).

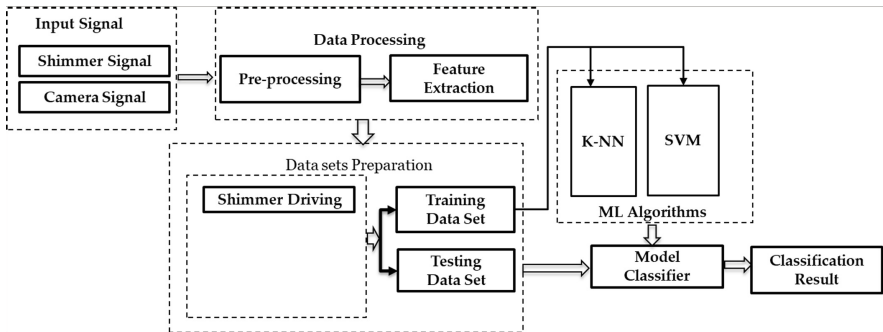


Fig. 1. Overview of the workflow for measuring CL

In the contact-based approach, a Shimmer GSR+ system was wired up to the subjects' body through a clamp to the earlobe, which provided the corresponding IBI values. The data recorded by the shimmer sensor was connected to a laptop via Bluetooth. The standard settings of Consensus were recorded as Unix_Timestamps, PPG_A13 as IBI and PPGtoHR. The IBI value was recorded by the shimmer system and saved in Excel sheets through a software called Consensus to monitor and manage the data gathered by the shimmersensing¹. The shimmer had a sampling rate of roughly 40000 after a

¹ <http://www.shimmersensing.com/products/consensus>.

5-min test run. Here the frequency of the sensor reading was 133 Hz. An overview of the workflow used in this study is presented in Figure, that describes the different parts i.e., Data collection, Data Processing, Data Set Preparation, Classification Learner, ML Algorithms, Model Classifier, Classification Results.

2.1 Data collection

This study includes 11 test subjects with age between 21 and 33 years. Among them 9 were men with average age of 26 and two were women with average age of 27. Again, four of them had lowest education in math, three of them had highest math education and the rest had average math courses. The mean value of how good they see themselves in mental arithmetic were 5, 9 in a scale of 10. The mean value of time spent playing digital games per week was 13 h, but the time variation is high between the test subjects. The one with the highest h per week had forty h, and the lowest had one hour Two of the eleven test subjects only played on mobile, and the rest played on console/computer.

The data was collected through a controlled study consists of seven phases [19], between phases there were a minor pause to set the next phase up and in the fifth phase there were a 15-min break. The *1st phase* of this experiment was a five-minute rest while sitting normal and watching TV silently. The *2nd Phase* was to complete the two different easy metal puzzles while sitting normal. Here, the test subject played an easy puzzle first, and when they have dissembled the first one, they started over the next one. When both the puzzles were dissembled the test subject would start to reassemble both the puzzles in the same order. This process was repeated for five minute. The *3rd Phase* was about to play both the metal puzzles with a higher difficulty. The *4th phase* was about the test subject rested for 15 min and were offered coffee and snacks. The *5th phase* was about to sit a normal watching TV silently same as in phase one, for five minute. In the *6th phase* the test subjects play a game ‘Mario Kart Double Dash’ on the 150cc course BOWER’S CASTLE competing against seven bots. The time of this phase was five minute; however, the time was varied depending on how well the test subjects performed. Finally, the *7th phase* is about to play the ‘Mario Kart Double Dash’ 150cc once again, but on RAINBOW ROAD, which is the final and the hardest course of the game. Here, instead of competing against seven bots the test subjects had to answer math questions while playing. The math questions had to be answered before the next question was asked, the questions were asked in every 30 s and five questions in total for each time. Table 1 illustrates the summary of the phases.

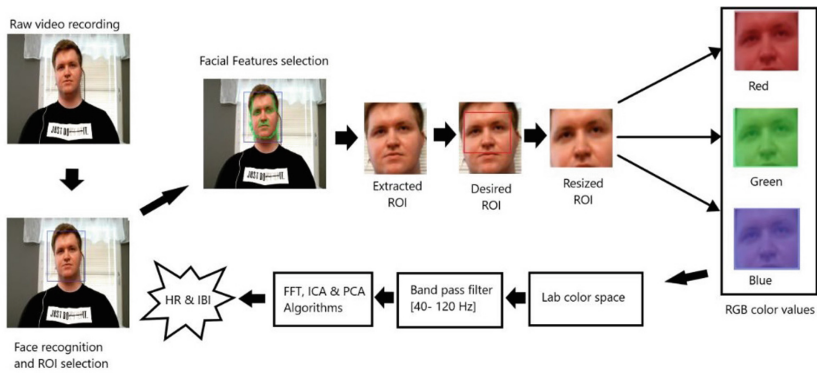
2.2 Data Pre-processing

For each phase of every test subject, the sample size was 40 000 on an average. A python program with the library OpenPyl was created in order to handle samples, the program both transformed the UNIX time to ‘date time’ and removed the samples where the IBI was – 1.0. Thus, a smaller sample size is reduced to 250–500 depending on the recorded time as well as the precision of the device. The reason of this huge decrease was that the IBI was calculated approximately once per second, since the HR was around 60–80 BPM for all the test subjects. This resulted in roughly 300 recorded IBI’s for each phase of every test subject, since the length of each experiment phase was five minute long.

Table 1. Summary of each phase during the data collection

Phases	1 st	2 nd	3 rd	4 th	5 th	6 th	7 th
Duration (min)	5	5	5	15	5	5	5
Description	Normal sitting and watching TV	Normal sitting and solving easy puzzle	Normal sitting and solving hard puzzle	Rest and drinking coffee/tea	Normal sitting and watching TV	Playing driving game	Playing driving game and solve math
Goal	Low cognitive level	Medium high cognitive level	High cognitive level	Relax and recover state	Low cognitive level	Medium high cognitive level	High cognitive level
Datasets	S0	S1	S2		S0	S1	S2

For the camera data, IBI was extracted from the recorded video and saved in a separate file. From each of the recorded video, first, the face of the test subject was detected and then a region of interest (ROI) was selected from the detected face [20]. Each ROI was converted into three color frames which are red, green and blue which are transferred into Lab color space or Lab signal. Three signal processing algorithm First Fourier Transform (FFT) [21], Independent Component Analysis (ICA) [22] and Principle Component Analysis (PCA) [23] were used to extract IBI from the Lab signal by applying a band pass filter of 40–120 Hz [24]. An overview of the camera system that extracts IBI from a video recording is presented in Fig. 2.

**Fig. 2.** Different steps of a camera system to extract IBI from camera data

2.3 Feature Extraction

Once the collected data were structured in a manner that would make it easier for a script to extract features from the recorded data the implementation of the feature extraction session started with the so-called trial and error phase in order to find a way of extracting all the HRV features properly. The purpose of the script was to reduce the total duration

of the recorded time into three min and only consider the middle of the recording and remove the data from the beginning and end of the recording to avoid faulty recording. These three min were then be further divided into three parts of one min each. Then for each min of these three min data, the feature function in MATLAB was used to calculate the 13 features where six time domain features and seven frequency domain features as shown in Table 2 and Table 3 respectively. Then the features for each phase was calculated for each test subject. This was resulted in a $3 * 14$ matrix linked to one class (in the controlled experiment phase), where SittingNormal = S0, Driving 1 or Puzzle 1 = S1, Driving 2 or Puzzle 2 = S2.

Table 2. List of different time-domain HRV features

Feature	Description	Unit
MeanNN	Is the average of all NN-intervals	ms
SDNN	Standard deviation of NN-intervals	ms
RMSSD	Root mean square successive RR interval differences	ms
SDSD	Standard deviation of successive differing between neighbor NN-intervals	ms
NN50	Number of pairs of successive NN-intervals differing by more than 50 ms	int
pNN50	Percentage of successive RR-intervals differing by more than 50 ms	%

Table 3. List of different frequency-domain HRV features

Feature	Description	Unit
VLF	Absolute power of very-low-frequency (0.003–0.04 Hz)	ms^2
LF	Absolute power of low-frequency (0.04–0.15 Hz)	ms^2
HF	Absolute power of high-frequency (0.15–0.40 Hz)	ms^2
TotalPower	Variance of all NN-intervals (approximately $\leq 0,4$ Hz)	ms^2
LF/HF ratio	Ratio of LF-HF power	%
LF (nu)	Relative power of low-frequency (0.04–0.15 Hz) in normal units	nu
HF (nu)	Relative power of high-frequency (0.15–0.40 Hz) in normal units	nu

2.4 Classification

For the classification using k-NN and SVM, the whole data sets were divided into four groups both for shimmer and camera data separately as follows:

- S0 vs S1, Normal sitting vs Driving normal
- S0 vs S2, Normal sitting vs Driving with distractions
- S1 vs S2, Driving normal vs Driving with distractions
- S0-S1-S2, Combined classification

First, 80% of the data for each group of data set was randomly selected for training and 20% for data was selected for testing. Binary classification was conducted for this paper works and two ML algorithms e.g. k-NN and SVM were considered an instance of supervised learning in which an algorithm learns to classify new observations from examples of labeled data [25]. SVM constructs a hyperplane or a set of hyperplane to analyses data to identify pattern and commonly used for classification and regression analysis [26]. K-NN is a nearest-neighbor classification model in which it can be altered both the distance metric and the number of nearest neighbors. Because a k-NN classifier stores training data, the model is used to compute re-substitution predictions. Alternatively, this model is used to classify new observations using the predict method [27].

Different kernel functions were investigated both for k-NN and SVM which were weighted, fine and medium kernel for k-NN and cubic, fine and medium kernel for SVM. The best cross-validation k-fold value was 10 for both k-NN and SVM. The binary SVM classifier was implemented according to [eq. 1], the `fitcsvm`² is a built-in function in MATLAB that returns a trained SVM model, based on the features of the training set x and the list with the corresponding class y . The parameters '*KernelFunction*' indicates that the kernel used in this model should be the variable k . The parameter '*KernelScale*' indicates that the kernel scale used in this model should be the variable s . '*BoxConstraint*' is the parameter indicating that the boxconstraint of the model should be the variable b .

$$trainedSVM = fitc(x, y', KernelFunction', k', kernelScale', s', BoxConstraint', b) \quad (1)$$

For k-NN model, [eq. 2] was implemented which is also a MATLAB built-in function. The function `fitck-NN` returns a trained K-NN model and based on the features of the training set x and the list with the corresponding class y . The parameter '*NumNeighbours*' flags that the variable k should be used as the number of neighbors for this K-NN model. '*DisatnceWeight*' is a parameter in the `fitc k-NN` that indicates that the variable w should be used as the distance weight.

$$trainedKNN = fitc(x, y', NumNeighbors', k, DistanceWeight', w) \quad (2)$$

The optimized classifiers were implemented for each training set of the class combinations based on the feature matrixes. The trained models were then given the features of the test sets and based on how they predicted the class of the features, the accuracy of each model could determine by how many correct class predictions it achieved. Each model executed 20 times for each data set and mean value of the result was considered.

3 Experimental Results

The classification learner revealed that the two best classification algorithms for the collected data were k-NN and SVM. Classification accuracy for the binary data sets (S0 vs S1), (S0 vs S2), (S1 vs S2) and (S0 vs S1andS2) and class classification accuracy for the combined data sets (S0, S1, S2) are calculated using ML algorithms i.e., k-NN and

² <https://se.mathworks.com/help/stats/fitcsvm.html>.

SVM. Also, F_1 -score is calculated for each of the data set using the ML algorithms. A summary of the classification accuracy and F_1 -score for the shimmer data is presented in Table 3 and Table 4 presents the classification accuracy for the camera data. The highest value of F_1 -score both for the shimmer and camera are highlighted in gray color in the table (Table 5).

Table 4. Classification results using Shimmer system

Classification	Data set	Shimmer			
		k-NN		SVM	
		Mean accuracy with 20 runs	F_1	Mean accuracy with 20 runs	F_1
Binary	S0 vs S1	0.74	0.7	0.74	0.78
	S0 vs S2	0.78	0.74	0.66	0.79
	S1 vs S2	0.58	0.41	0.58	0.65
Combination	S0 vs S1 and S2	0.72	0.86	0.68	0.85

Table 5. Classification results using Camera system

Classification	Data set	Camera			
		k-NN		SVM	
		Mean accuracy with 20 runs	F_1	Mean accuracy with 20 runs	F_1
Binary	S0 vs S1	0.71	0.71	0.70	0.68
	S0 vs S2	0.81	0.84	0.74	0.74
	S1 vs S2	0.67	0.57	0.65	0.66
Combination	S0 vs S1 and S2	0.79	0.87	0.72	0.88

For k-NN and SVM, a graphical representation of average F_1 -score between shimmer and camera is presented in Fig. 3 and Fig. 4. From the figures it can be seen that F_1 -score for camera is larger than Shimmer both for k-NN and SVM considering binary and combined class classification.

4 Discussion and Conclusion

The experimental result shows that highest accuracy has been achieved in a controlled environment using the camera recording during the driving phase. Also, this shows that binary classification using the k-NN algorithm with the class combination of S0 vs S2 achieves an accuracy of 81% which was higher than the expected accuracy from the classification learner by one percent. With the same class combination and setup

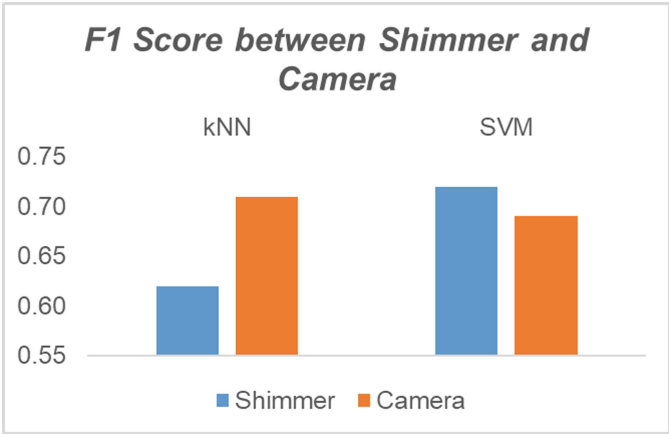


Fig. 3. Graphical representation of F_1 -score between Shimmer and Camera data considering binary classification.

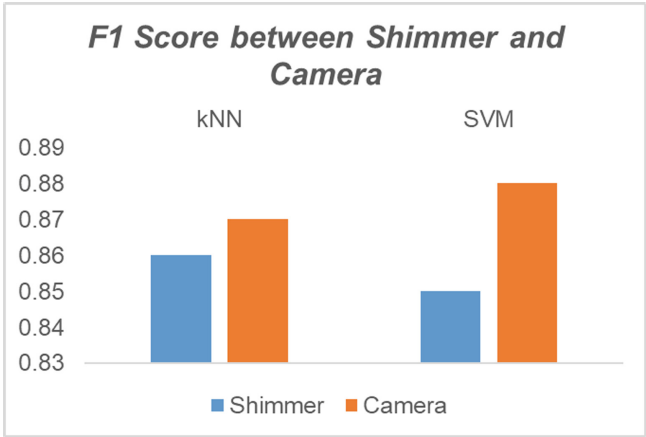


Fig. 4. Graphical representation of F_1 -score between Shimmer and Camera data considering classification of combined class

recorded with the Shimmer sensor system only got 78% accuracy, which is higher than the expected accuracy 70%. For the class combination S0 vs (S1 and S2), the highest F_1 score 84% was achieved with medium gaussian SVM using the feature matrix recorded from the Camera system.

The results also demonstrates that the ML algorithms could classify CL in Driving using video games with equal or even higher accuracy than a traditional way (i.e., using metal puzzles). This is significant because controlled experiments that involve actual video games have not been extensively used in the state-of-the-art research. This research also reveals that the accuracy in classifying CL between driving normal and driving with distractions was greater than the accuracy between the Easy puzzle and Hard puzzle.

The reason for this might have been that the difficulty between the easy puzzle and hard puzzle was too small. If math questions had been added to the hard puzzle phase of the controlled experiment the results could have been different.

The best improvement was achieved from the testing with an increase of the accuracy with 3% based on the focus to feature selection. The MATLAB function `sequentialfs` (Sequential Feature selection) was used to determine the best features among the extracted features. This shows that the NFFT feature, was not necessary and therefore NFFT was removed from the feature sets. After analyzing more into the features, the MeanNN, RMSSD, VLF and TotalPower were found the most robust features while considering stable accuracies. The data set S2 is provided more accuracy compare to S0 using contact-free approach since it was the most difficult and the hardest difficulty, the driver usually ended up so far behind the bots on the course that they were not disturbed by the bots in the game. The math questions were used to make distraction of the driver which means there should have difference level of cognitive load; however, practically it was not found the difference between cognitive load levels due to less distraction of the drivers by math questions. The choice of an easier track could have made it possible to have the data differentiate even more than it did. Also, the time determined when a math question would be asked, every 30 s. This made most of the questions to be asked when the driver had a straight line without any worries and could therefore answer the question more easily. Also, more questions could have been held at hand in order for the questions to stretch across the whole course.

When conducting the controlled experiment, the two recording systems (i.e., Shimmer and Camera) had to be started at the same time. The challenge with this was that a button had to be pressed on each system in order to start the recordings. This made that the two system recordings had a minor time difference of a few ms for each recording for each of the test subject. Also, when cutting the recordings from five minute into three minute the implementation cut at the one-minute mark. The same thing occurred when the implementation cut the last minute from the recording at the three-minute mark. Because of this cutting the time gap was small enough that it should not affect the result.

With these findings it seems plausible that the contact-free system can be used reliably in real driving scenario. However, it would also be interesting to investigate if there is something that increase the CL more than other things and try to categorize how much it increased the CL. Also, do more experiments with the game over a longer period of time, to see how different test subjects will adapt to the Mario Kart game. Here, only the frequency and the time-domain features were considered. So, it would also be interesting to involve the non-linear HRV features in the future.

Acknowledgment. The authors would like to acknowledge Embedded Sensor Systems for Health Plus (ESS-H+) for their support of the research projects. The authors would also like to acknowledge all the participants for supporting in this study.

References

1. Kim, J.-P., Lee, S.-W.: Time domain EEG analysis for evaluating the effects of driver's mental work load during simulated driving, pp. 79–80 (2017)

2. Miyaji, M., Kawanaka, H., Oguri, K.: Study on effect of adding pupil diameter as recognition features for driver's cognitive distraction detection. In: 2010 7th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP 2010), pp. 406–411 (2010)
3. Road Safety Annual Report 2019. OECD Publishing, Paris, OECD/ITF (2019)
4. Sena, P., d'Amore, M., Pappalardo, M., Pellegrino, A., Fiorentino, A., Villecco, F.: Studying the influence of cognitive load on driver's performances by a Fuzzy analysis of Lane Keeping in a drive simulation. IFAC Proc. **46**(21), 151–156 (2013)
5. Palinko, O., Kun, A.L., Shyrokov, A., Heeman, P.: Estimating cognitive load using remote eye tracking in a driving simulator. Presented at the Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, Austin, Texas (2010). <https://doi.org/10.1145/1743666.1743701>
6. Kumar, N., Kumar, J.: Measurement of cognitive load in HCI systems using EEG power spectrum: an experimental study. Procedia Comput. Sci. **84**, 70–78 (2016)
7. Reimer, B., Mehler, B., Coughlin, J., Godfrey, K., Tan, C.: An on-road assessment of the impact of cognitive workload on physiological arousal in young adult drivers, pp. 115–118 (2009)
8. Zander, T.O.: Evaluation of a dry eeg system for application of passive brain-computer interfaces in autonomous driving. Front. Hum. Neurosci. **11**, 78 (2017). (in English)
9. Rahman, H., Ahmed, M.U., Begum, S.: Vision-based remote heart rate variability monitoring using camera. In: Ahmed, M.U., Begum, S., Bastel, J.-B. (eds.) HealthyIoT 2017. LNICTS, vol. 225, pp. 10–18. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-76213-5_2
10. McDuff, D., Gontarek, S., Picard, R.: Remote measurement of cognitive stress via heart rate variability. In: 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014, vol. 2014, pp. 2957–2960 (2014)
11. Sahadat, M.N., Consul-Pacareu, S., Morshed, B.I.: Wireless ambulatory ECG signal capture for HRV and cognitive load study using the NeuroMonitor platform. In: 2013 6th International IEEE/EMBS Conference on Neural Engineering (NER), pp. 497–500 (2013)
12. Shaffer, F., Ginsberg, J.P.: An overview of heart rate variability metrics and norms. Front. Pub. Health **5**, 258 (2017). (in English)
13. Hussain, S., Chen, S., Calvo, R., Chen, F.: Classification of cognitive load from task performance & multichannel physiology during affective changes (2007)
14. Solovey, E.T., Zec, M., Garcia Perez, E.A., Reimer, B., Mehler, B.: Classifying driver workload using physiological and driving performance data: two field studies. In: Conference on Human Factors in Computing Systems – Proceedings (2014)
15. Engström, J., Johansson, E., Östlund, J.: Effects of visual and cognitive load in real and simulated motorway driving. Transp. Res. Part F: Traffic Psychol. Behav. **8**(2), 97–120 (2005)
16. Reimer, B., Mehler, B.: The impact of cognitive workload on physiological arousal in young adult drivers: a field study and simulation validation. Ergonomics **54**, 932–942 (2011)
17. Liu, A., Chen, C.: Towards practical driver cognitive load detection based on visual attention information. Masters of Science, University of Toronto, Canada (2017)
18. Wang, C., Guo, J.: A data-driven framework for learners' cognitive load detection using ECG-PPG physiological feature fusion and XGBoost classification. Procedia Comput. Sci. **147**, 338–348 (2019)
19. Sörman, J., Gestlöf, R.: Contact-free cognitive load classification based on psycho-physiological parameters. Bachelor of Science, Mälardalen University, Sweden (2019)
20. Rahman, H., Begum, S., Ahmed, M.U., Funk, P.: Real time heart rate monitoring from facial RGB color video using Webcam. In: 29th Annual Workshop of the Swedish Artificial Intelligence Society (SAIS) 2016, Malmö, Sweden (2016)

21. Buijs, H., Pomerleau, A., Fournier, M., Tam, W.: Implementation of a fast Fourier transform (FFT) for image processing applications. *IEEE Trans. Acoust. Speech Signal Process.* **22**(6), 420–424 (1974)
22. Comon, P.: Independent component analysis, a new concept. *Signal Process.* **36**, 287–314 (1994)
23. Kramer, M.A.: Nonlinear principal component analysis using autoassociative neural networks. *AIChE J.* **37**(2), 233–243 (1991)
24. Rahman, H., Ahmed, M.U., Begum, S.: Non-contact heart rate monitoring using lab color space. In: 13th International Conference on Wearable, Micro & Nano Technologies for Personalized Health (PHealth 2016), Crete, Greece, 29–31 May 2016 (2016)
25. Alpaydin, E.: Introduction to Machine Learning. MIT Press (2010)
26. Cortes, C., Vapnik, V.N.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
27. Tamrakar, P., Roy, S.S., Satapathy, B. Ibrahim, S.P.S.: Integration of lazy learning associative classification with k-NN algorithm. In: 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), pp. 1–4 (2019)