Accident Analysis and Prevention 120 (2018) 188–194

Contents lists available at ScienceDirect

Accident Analysis and Prevention

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

Modular design of fatigue detection in naturalistic driving environments

Hilal Al-libawy[a,b,*], Ali Al-Ataby[a], Waleed Al-Nuaimy[a], Majid A. Al-Taee[a]

[a] Department of Electrical Engineering and Electronics, University of Liverpool, Brownlow Hill, Liverpool L69 3GJ, UK
[b] Department of Electrical Engineering, University of Babylon, Iraq

ARTICLE INFO

Keywords:
Bayesian combiner
Driver fatigue
Machine learning
Modular structure
Performance optimisation
Particle swarm optimisation

ABSTRACT

Research in driver mental fatigue is motivated by the fact that errors made by drivers often have life-threatening consequences. This paper proposes a new modular design approach for the early detection of driver fatigue system taking into account optimisation of system performance using particle swarm optimisation (PSO). The proposed system is designed and implemented using an existing dataset that was simultaneously collected from participants and vehicles in a naturalistic environment. Four types of data are considered as fatigue-related metrics including: vehicle acceleration, vehicle rotation pattern, driver's head position and driver's head rotation. The driver's blink rate data is used in this work as a proxy for ground truth for the classification algorithm. The collected data elements are initially fed to input modules represented by ternary neural network classifiers that estimates alertness. A Bayesian algorithm with PSO is then used to combine and optimise detection performance based on the number of existing input modules as well as their output states. Performance of the developed fatigue-detection system is assessed experimentally with a small data samples of driver trips. The obtained results are found in agreement with the state-of-the-art in terms of accuracy (90.4%), sensitivity (92.6%) and specificity (90.7%). These results are achieved with significant design flexibility and robustness against partial loss of input data source(s). However, due to small sample size of dataset ($N = 3$), a larger dataset need to be tested with the same system framework to generalise the findings of this work.

1. Introduction

Reliable and robust driver fatigue detection systems are becoming essential requirements for road safety due to the dangerous and often fatal consequences of road accidents caused by fatigued drivers. The onset of mental fatigue is usually accompanied by slow reaction time, impaired judgement and may ultimately lead to falling asleep behind the steering wheel. Road accidents caused by fatigued drivers can have fatal and devastation consequences (Centre for Road Safety, 2016). Numerous driver fatigue symptoms have been reported in the literature along with relevant systems used to detect them (Abbood et al., 2014; Stork et al., 2015), with varying degree of success.

Different swarm optimisation methods have been proposed in the transportation sector to find the optimal traffic network situation (Sharma and Kumari, 2015; Sandberg and Wahde, 2008). However, only few studies have identified the use of optimisation techniques to enhance the performance level of the modular organisation (Durán et al., 2012). Despite the important contributions reported in these studies and others, identifying a practical, robust and flexible fatigue-detection approach remains an elusive goal.

In this work, a scalable modular design approach is considered to build a system using a Bayesian combiner and a particle swarm optimiser (PSO). This enables the utilisation of input modules depending on availability. The Bayesian combiner, which improves the detection accuracy with the aid of PSO, deals with the number of existing input modules as well as their states (i.e. alert, mild fatigue and fatigued). Unlike existing systems, this makes the proposed system flexible in terms of the available input data and more robust against losing one or more data sources.

The paper is organised as follows. A theoretical background on modularity, Bayesian algorithm and PSO is presented in Section 3. Section 4 details the study dataset. The overview for the proposed system architectures and methodology are described in Section 5. The obtained results are presented and discussed in Section 6. Finally, the work is concluded in Section 7.

2. Related work

Mental fatigue evaluation has been reported in a numerous amount of research articles (Meiring and Myburgh, 2015; Al-Libawy et al., 2016a, 2016b, 2017). The available driver fatigue-detection literature has agreed to categorise fatigue-related symptoms into three categories

* Corresponding author at: Department of Electrical Engineering and Electronics, University of Liverpool, Brownlow Hill, Liverpool L69 3GJ, UK.
E-mail address: eng.hilal_al-libawy@uobabylon.edu.iq (H. Al-libawy).

https://doi.org/10.1016/j.aap.2018.08.012
Received 11 July 2017; Received in revised form 3 March 2018; Accepted 13 August 2018
0001-4575/

based on either generation or detection perspectives. The first category includes the biological symptoms which capture the development of driver fatigue using metrics such as heart rate, skin temperature and skin conductivity (Al-Libawy et al., 2015). Different detection techniques and devices are used to detect the biological signs of fatigue such as wearable devices or sensors built into the vehicle (Zhang, et al., 2017). The second is the behavioural category which is mainly noticed from driving style (Engelbrecht et al., 2015). Metrics such as steering wheel angle, acceleration pattern, lateral movements and braking pattern are the main measures used to quantify fatigue status of the driver (Meiring and Myburgh, 2015; Li et al., 2017). The last category is visual signs of drivers, such as facial features, percentage of eye closure, blink rate, yawn rate and others (Sigari et al., 2014).

More recent works of fatigue-detection systems have used more than one fatigue-related metric to improve the detection accuracy (Stork et al., 2015). The detection system fuses and combine the calculated metrics in three main levels. The first is the raw or filtered data level which is used when different sensors measure the same metric (Koenig et al., 2015). The second is the features level and can detect fatigue status from features even when they are extracted from different types of sensors (Yin et al., 2016). The last level is the abstract or the decision level which combines multi-module outputs to calculate an enhanced and accurate output. Several integrated driver fatigued detection systems have been developed that combine different classifiers decisions using different fatigue-related metrics (Craye et al., 2016). Despite the good results obtained from simulated environments, the robustness of these decision-combining systems was not tested under condition of partial loss of input data.

While several experiments were carried out to measure and quantify driver behaviour in real environments, only a small portion of these was focused on driver fatigue in naturalistic environments (Fu et al., 2016; Li et al., 2017). The work reported in Fu et al. (2016) adopted three fatigue-related physiological metrics (EEG, EMG and respiration rate), but the method for measuring these metrics is not very practical in real environments. The results in publication Li et al. (2017) are calculated based on one metric (steering wheel angle) which may not be available in many vehicles and require third party devices to be installed.

3. Theoretical background

This section presents a theoretical background for three key aspects of the proposed work: the importance of modularity, Bayesian combiner and PSO.

3.1. Importance of modularity

The need for a system that can combine heterogeneous subsystems (modules) efficiently is behind the idea of *"interdependence within and independence across modules"* (Hatch, 2001) which is one of the definitions of modularity. Modular design is preferred over the integral design due to its benefits including (Avigad, 2016; Pradhan et al., 2011):

(a) *Reliability and robustness*. The modular structure enhances the robustness and improves the reliability of certain system if they are designed properly to maintain their functional mission even if they lose one or more of their modules.
(b) *Flexibility*. The need for a flexible system is an essential practical requirement (Sanchez and Mahoney, 1996). For a fatigue-detection system where a variety of detection methods are available, it is very important to design the system to be flexible to accommodate a variety of configurations for different working environments.
(c) *Comprehensibility*. In complex systems such as fatigue-detection systems, modular structure makes them more understandable and easier to handle on a module level.
(d) *Independence*. A primary motivation of using a modular structure is the independence of each individual module from other modules. This feature is very helpful for modular systems to integrate heterogeneous modules together without requiring agreement on internal details.
(e) *Abstraction level*. This level is responsible for the interfacing between modules. The abstraction and independence of the modules help to make the system more practical and flexible.

3.2. Bayesian combiner

The modular structure system needs a frame to be plugged in to produce the combined final output. Several combination algorithms have been proposed to combine heterogeneous sets of modules (e.g. majority voting, weighted majority voting, or Bayesian combiner) (Bahler and Navarro, 2000; Kuncheva, 2004). The Bayesian combiner is ideally suited for problems when the output of the modules is independent even when the number of modules is dropped to two (Kim and Ghahramani, 2012).

The Bayesian combiner works at the abstraction level of the output of L modules, each module M_i predicts class label $b_i \in \boldsymbol{H}$, $i = 1, \ldots, L$. So, any input set $\boldsymbol{x} \in \boldsymbol{R^n}$, can be combined, the L module outputs produce a vector $\boldsymbol{b} = [b_1, \ldots, b_L]^T \in \boldsymbol{H^L}$. Bayes' theorem Vapnik and Vapnik (1998) of probability computes the posterior probability of module M_i based on the prior probability $P(h_j)$ where h_j is the actual class label ($h_j \in \boldsymbol{H}$, $j = 1, \ldots, C$ and C is the number of classes); as well as based on the likelihood $P(\boldsymbol{b}|h_j)$ of the evidence $\boldsymbol{b}$. The independence assumption is maintained and allows the conditional probability of the module M_i labels the input $\boldsymbol{x}$ in class $b_i \in \boldsymbol{H}$ to be presented as follows:

$$P(\boldsymbol{b}|h_j) = P(b_1, b_2, \ldots, b_L|h_j) = \prod_{i=1}^{L} P(b_i|h_j) \tag{1}$$

Bayes' rule can be described mathematically as follows:

$$P(h_j|\boldsymbol{b}) = \frac{P(\boldsymbol{b}|_j)P(h_j)}{P(\boldsymbol{b})} \tag{2}$$

3.3. Particle swarm optimisation (PSO)

Particle swarm optimisation is a meta-heuristic algorithm devised by Kennedy and Eberhart in 1995 Engelbrecht (2006) which is inspired by the social behaviour of birds. Several versions of PSO were derived later to cover new applications or to address some limitations and challenges discovered with the original version (Poli et al., 2007).

PSO can find the optimal solution for an optimisation problem in a D-dimensional hyperspace. A swarm of N particles is recruited to find the best position according to the individual perspective (*Pbest*) and the overall perspective (*Gbest*) (Lazinica, 2009). Each particle tries to update its position (solution) to achieve the best fitness value and minimise the cost function. The update stochastic function is based on three parts: inertia part, self-knowledge part and team-work part. The update rule is determined as follows:

$$v_i^{k+1} = \mathrm{w}v_i^k + c_1R_1 \times (\mathrm{Pbest}_i^k - x_i^k) + c_2R_2 \times (\mathrm{Gbest}_i^k - x_i^k) \tag{3}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{4}$$

where w is the inertia (habitual behaviour) weight, c_1 is an acceleration constant of the self-knowledge (memory) component, c_2 is an acceleration constant of the team-work component, and R_1 and R_2 are random numbers.

PSO is used in this work to improve the performance of the Bayesian combiner. The cost function of the optimiser is a function of the combiner accuracy, while the best solution will be represented in the best set of weights related to each module (it is not related to the inertia weight of the optimiser update rule in Eq. (3)).