

Non-intrusive Driver Drowsiness Detection based on Face and Eye Tracking

Ameen Aliu Bamidele¹, Kamilia Kamardin², Nur Syazarin Natasha Abd Aziz³, Suriani Mohd Sam⁴
Irfanuddin Shafi Ahmed⁵, Azizul Azizan⁶, Nurul Aini Bani⁷, Hazilah Mad Kaidi⁸

Razak School of Technology and Informatics, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia^{1, 3, 4, 5, 6, 7, 8}
Malaysia-Japan International Institute of Technology, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia²
Wireless Communication Centre, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia^{2, 6, 8}

Abstract—The rate of annual road accidents attributed to drowsy driving are significantly high. Due to this, researchers have proposed several methods aimed at detecting drivers' drowsiness. These methods include subjective, physiological, behavioral, vehicle-based, and hybrid methods. However, recent reports on road safety are still indicating drowsy driving as a major cause of road accidents. This is plausible because the current driver drowsiness detection (DDD) solutions are either intrusive or expensive, thus hindering their ubiquitous nature. This research serves to bridge this gap by providing a test-bed for achieving a non-intrusive and low-cost DDD solution. A behavioral DDD solution is proposed based on tracking the face and eye state of the driver. The aim is to make this research an inception to DDD pervasiveness. To achieve this, National Tsing Hua University (NTHU) Computer Vision Lab's driver drowsiness detection video dataset was utilized. Several video and image processing operations were performed on the videos so as to detect the drivers' eye state. From the eye states, three important drowsiness features were extracted: percentage of eyelid closure (PERCLOS), blink frequency (BF), and Maximum Closure Duration (MCD) of the eyes. These features were then fed as inputs into several machine learning models for drowsiness classification. Models from the K-nearest Neighbors (KNN), Support Vector Machine (SVM), Logistic Regression, and Artificial Neural Networks (ANN) machine learning algorithms were experimented. These models were evaluated by calculating their accuracy, sensitivity, specificity, miss rate, and false alarm rate values. Although these five metrics were evaluated, the focus was more on getting optimal accuracies and miss rates. The result shows that the best models were a KNN model when $k = 31$ and an ANN model that used an Adadelta optimizer with 3 hidden layer network of 3, 27, and 9 neurons respective. The KNN model obtained an accuracy of 72.25% with a miss rate of 16.67%, while the ANN model obtained 71.61% and 14.44% accuracy and miss rate respectively.

Keywords—Driver Drowsiness Detection (DDD); face tracking; eye tracking; K-nearest Neighbors (KNN); Support Vector Machine (SVM); Logistic Regression; Artificial Neural Networks (ANN)

I. INTRODUCTION

Drowsy driving is considered one of the major contributors to road accidents all over the world. Every year, thousands of deaths or severe injuries are recorded due to drivers falling asleep while driving [1]. According to a report by the National Highway Traffic Safety Administration (NHTSA), driver drowsiness accounts for approximately 83,000 crashes, 37,000 injuries, and 900 deaths in the United States alone [2]. Another

recent report by the World Health Organization (WHO) on road safety reveals that approximately 1.2 million road deaths occur annually from 2001 to 2013 [3]. Including the aforementioned, many other reports also identified road accidents as a rising cause of human deaths and a significant amount of these accidents were attributed to driver drowsiness.

Due to this severity, several types of research works have been conducted to prevent drivers from getting drowsy while driving. One of the solutions employed involved educating the drivers on the adversities of driving drowsy. This passive approach entails awareness of the effects of a distorted sleeping schedule, sleep deprivation, and the merits of getting a good night sleep [4]. Although these are requisites to completely eradicate drowsy driving [5], and could probably keep drivers alert at specific times, they are, however, not capable of determining whether a driver would remain alert in extreme situations like driving for a long period of time, not to mention, of proffering a solution if drowsiness occurs unexpectedly. Thus, an approach capable of measuring driver drowsiness in real-time is required. Moreover, there are drivers experiencing distorted sleep schedules due to the nature of their job, or with sleep disorders [6], who could still fall asleep even after adequate sleep.

However, as expected, numerous real-time approaches, providing on-board monitoring of drivers' drowsiness state have also been developed. This is, in fact, the focus of this problem domain. Techniques including physiological, behavioral, vehicle-based, and hybrid methods have been proposed and implemented. Among these techniques, physiological approaches are regarded as the most accurate driver drowsiness detection (DDD) method [7] with electroencephalogram (EEG) and electrocardiogram (ECG) being the most exploited approaches. EEG and ECG are used for tracking the driver's brain activity [8-10] and heart pulse rate [11-13] respectively. Although both approaches offer higher accuracy, they are actually not feasible in the real-world [14]. This is because they require electrodes being attached to the driver, which could cause discomfort as well as distractions [15, 16]. This would likely trace back to the initial problem they intended to prevent or solve.

Behavioral and vehicle-based methods, on the other hand, are non-intrusive. Even though they attain slightly lesser accuracy than their physiological counterparts, they have received significant interest from researchers. This is plausible

due to their portability into real-world scenarios. Moreover, according to [5], current researches on these two methods signify their accuracy is close to that of physiological methods. An example is the tracking of drivers' steering wheel movements (SWM) by [17], which obtained a reasonable accuracy. Authors in [15, 16] also used cameras and computer vision techniques to track drivers' facial and eye features as a means of detecting driver drowsiness.

This project, however, focuses on exploring DDD by tracking drivers' face and eye states. This approach was selected not only because it is one of the most promising techniques for accurately detecting driver drowsiness [15], but also because it is a potential solution for pragmatically achieving DDD ubiquity in the real world. The major hindrance of this approach, however, is its current high cost and quality of camera requirement. While SWM is a promising approach with low-cost application possibilities, SWM and other vehicle-based methods have potential for generating high false alarm rates [5], leading to more computational requirements.

This project aims to exploit face and eye tracking DDD approach as a test-bed for achieving a low-cost and non-intrusive DDD solution. To achieve this, a face and eye tracking model and several machine learning models were developed. The face and eye tracking model were used to detect if the driver's eyes are opened or closed. This tracking was done on a video dataset containing almost 2 hours of driver drowsiness states occurring during the day and night. The night data were very crucial in generalizing the accuracy of the machine learning models as drowsiness is predicted to occur mostly at night [18, 19]. Once the eye states were captured, they were passed to the machine learning models for classification.

Almost all DDD approaches use machine learning for classifying their drowsiness state. However, most of these approaches are usually limited to two or three models when evaluating their accuracy. With the driver's eye state data, over twenty models were developed employing K-Nearest Neighbors (KNN), Support Vectors Machine (SVM), Logistic Regression, and Artificial Neural Networks (ANN) classifiers for classifying the eye states into awake or drowsy. With this experiment, contribution of a face and eye state model was possible with a robust machine learning classification comparison for the face and eye tracking DDD domain.

So far, the paper vaguely touched on what Driver Drowsiness Detection is, while highlighting how the task at hand can be accomplished. But, in order to get to the crux of the problem, there is a need to understand the problem first, while focusing on the history and statistical study of the problem. This will potentially alleviate the concern caught on a developer's radar, to urge a solution at the earliest. The following section caters to these objectives coherently.

II. PROBLEM BACKGROUND

Road safety is a significant issue for many countries and many safety-related organizations. As population and urbanization grows, more vehicles and road users are expected on the highway networks. Thus, this leads to increased road

management demand. The current upsurge in these factors (population and urbanization) has, however, stigmatized the current road management techniques. Recent reports are signaling the implications of poor or insufficient road management, yielding high occurrence of road accidents. As shown in Fig. 1 and 2, road accidents account for approximately 1.2 million deaths since the last decade [3]. These deaths were identified as the major causes of youth demise. Another report from Malaysia Institute of Road Safety Research (MIROS) indicates a total of 476,196 road crashes occurred in 2014 with 6,674 and 4,432 leading to deaths and serious injuries, respectively [20].

Different factors contribute to road accidents. However, the most prominent are those related to impairing the driver's response to road tasks. These factors, which include alcohol impairment, drugs usage, aging, distraction, and drowsiness, are responsible for 31% of fatal accidents in the US according to FARS [21]. Drowsy driving, as a result of these factors, has received significant attention as a major root cause of road crashes. According to the National Sleep Foundation (NSF), 37% of drivers in the US are guilty of drowsy driving and 23% have done it once in every month [22]. Another report by [23] revealed that 16.5% of lethal crashes and 12.5% of accidents, requiring admission of road users to hospitals, are as a result of drowsy driving.

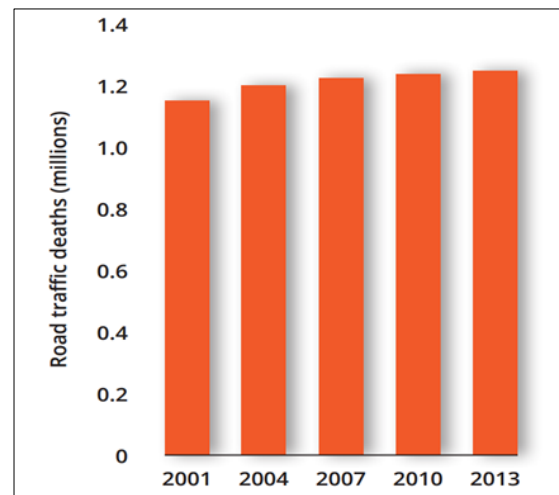


Fig. 1. Number of Road Deaths, Worldwide [3].

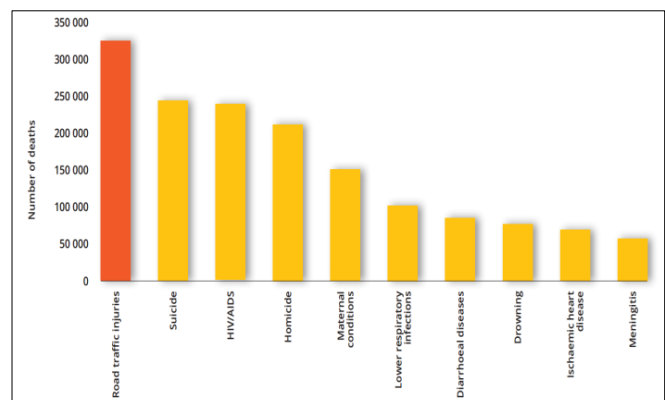


Fig. 2. Top Ten Causes of Death among People Aged 15-29 Years, 2012 [3].

Drowsy driving is considered a subset of impairment-related conditions that hinder the driver's response to required road tasks at the appropriate time. This state can arise either due to sleep disorders, sleep deprivation, or driving for a long period of time [6, 24]. When this occurs, the driver's driving performance, expertise, and decision making are adversely affected [16], thus leading to higher chances of accident occurrence. Although if the driver is not alone, he/she might be alerted by a passenger, however, this is not usually the case as most drowsiness-related crashes occur when the driver is alone [25].

Drowsy driving is a critical issue as its adversities do not only affect the driver but is also a threat to all other road users in the society. Several solutions, employing different approaches, to curb this problem have been developed and deployed. They include: subjective, physiological, behavioral, vehicle-based, and hybrid solutions [26-27]. Apart from subjective methods, which are questionnaire-based, and hybrid methods, which are just a combination of other methods, the other three methods are at the core of the DDD research domain.

Physiological methods, including EEG, ECG, and electrooculogram (EOG), are regarded as the most accurate DDD approaches [14-15,17,28-29]. A major contribution to their high accuracy is their capability of gaining direct access to drivers' inner body electrical signals, which can, however, be used for immediate detection of alertness changes [30]. Nevertheless, these methods are mostly restricted to lab environments [26]. This is plausible because they are intrusive in nature, and can cause discomfort or distraction to the driver. Vehicle-based and behavioral methods, on the other hand, are non-intrusive [14]. Although their accuracy is slightly lesser than their physiological counterpart, a recent review by [5] indicates that their accuracy is catching up. The major issue with these methods, however, is the high cost of equipment required.

Within the DDD vehicle-based solutions ecosystem, Lane deviation [24,31] and SWM [17] approaches have received ample research significance, yielding adequate accuracies. Of these two approaches, SWM poses the utmost potential and has even been experimented on low-cost applications. However, most vehicle-based approaches are subjective to false alarms because they are constrained by vehicle type, driver's experience, road condition, etc. Author in [15] abates their accuracy because of the extra computation required.

Behavioral methods tend to bridge this gap (that is the excessive computation and intrusiveness) by offering competitive accuracy with tolerable computational expenses. Researchers targeting this method usually scope their driver behavioral definition to the drivers' facial features, since this is the most definitive physical part of the body that depicts a drowsiness state. As one of the most researched DDD method, several optimizations and comparisons have been made against its approaches. Percentage of eyelid closure (PERCLOS), a popular behavioral approach, is usually used as a benchmark against other DDD solutions' accuracy. Apart from PERCLOS, yawning detection and head pose tracking are also prominent approaches within this domain. However, there is still a need

for a low-cost implementation of this approach. This is because previous works only employ expensive and high-quality cameras.

The current statistics, relating to road accidents in conjunction with those attributed to driver drowsiness, indicate there are still significant adversities on road highways. Although there is notably vast research on DDD, the impact of these researches in the real world appears to be dawdling. This is because only a few solutions are actually deployed into real-world scenarios. Presently, DDD solutions fall into one or more of these categories:

- Intrusive; thus, not feasible in the real-world.
- Very expensive; therefore, only available to minute fraction of road users.
- Affordable; but, lacking robustness.

A prospective method for mitigating the above mentioned is to perform DDD through the face and eye tracking. However, this method also has its own share of constraints, which are tied to the cost and quality of cameras required. Nevertheless, these constraints can be triumphed by channeling most of the required processes to the computational side of the system.

In order to achieve a solution balancing intrusiveness, affordability and accuracy, this project seeks to develop a face and eye tracking DDD model and evaluate it using vast classes of prominent machine learning classifiers. With this, their accuracies can be used for selecting the appropriate model for a device with constrained specifications. Thus, the aim is to find a solution to the problem of how driver drowsiness can be detected effectively by the means of a solution deployable in the real-world which balances affordability, intrusiveness, and accuracy.

III. LITERATURE REVIEW

Drowsiness refers to an awake state whereby there is an excessive urge or tendency to fall asleep. Whenever this occurs, the correspondent is usually afflicted by severe lethargy, and mental weakness, thus leading to depleted performance. Also, while it is common to find drowsiness being interchanged with fatigue in literature, both states are actually different concepts [32-33]. Fatigue, on the other hand, implies an extreme tiredness state which could be caused by several means including drowsiness or physical activities like exercises. Thus, drowsiness is a function of fatigue.

Drowsiness, also known as sleepiness, can emerge depending on several factors including being awake for a long period of time, sleep disorders, and sleep schedule distortion or medications [34]. Cases of sleep disorders include hypersomnia, sleep apnea syndrome, circadian rhythm sleep disorders, etc. [6] while sleep schedule distortion involves staying awake, or working for a long period of time, mostly due to working night shifts or driving through a long journey. Medications, on the other hand, involve using tranquilizers or sleeping pills [34].

The effect of drowsiness is significant in all domains. The cost of these effects, however, varies across each domain. Within the transportation domain, many reports have identified

drowsiness as a significant cause of road accidents leading to severe injuries and deaths [2-3]. A survey conducted by [6], seeking to correlate traffic accidents resulting from drowsy driving and the quantity of sleep the drivers had before the accident, revealed that 26.3% of accidents, in their sample, were caused by drivers that had less than 6 hours of sleep, the day before the accident. This implies that these drivers have higher chances of becoming drowsy while driving. Another finding by [35] led to a consensus that drivers who have slept for less than 2 hours within their past 24 hours are not competent to drive, and those who had slept for 3 to 5 hours have high chances of being impaired while driving.

All these issues concerning drowsiness have incited researchers to find a way for detecting and managing drowsiness. Managing drowsiness (not within the scope of this research) in and of itself is not an easy task [32], and so is its detection. Detecting drowsiness can be as simple as just looking at the correspondent's behavioral changes. However, this becomes a difficult task when the individual is alone [36]. Moreover, most of the drowsy driving crashes occur when the driver is alone [37]. Furthermore, DDD is known to be a challenging task because there's no standardized measure for detecting how drowsy the driver was or at a specific period of time [35,37]. Up till today, the current driver's drowsiness detection still suffers from either of varying drivers' characteristics, road environment or the vehicle type [14-15,38].

Although detecting drowsiness is not an easy task, several approaches have been devised to facilitate its detection. These approaches are usually categorized into either of the technological or non-technological [17], intrusive or non-intrusive [14,39], and objective or subjective approaches [40-41]. Within the driving context, these categories are mapped to five methods, which are subjective, physiological, behavioral, vehicle-based, and hybrid methods [42]. This mapping is shown in Table I.

A. Subjective Methods

Subjective methods involve assessing the drivers' current level of drowsiness by subjecting them to ratings in the form of questionnaires. These ratings are usually self-evaluated [40] or evaluated by experts watching the driver in action [15]. To detect the changes in a driver's drowsiness state, [14] conducted a pre-experimental, mid-experimental, and post-experimental Karolinska Sleepiness Scale (KSS) exercise. These ratings were the keys to define their drowsiness ground truth. [15], on the other hand, employed three experts to evaluate the drivers' state and a Johns Drowsiness Scale (JDS) was used. Other methods like the Stanford Sleepiness Scale (SSS), as shown in Table II, and Epworth Sleepiness Scale has also been applied by researchers.

As insinuated above, subjective methods are usually not used distinctively. They are used alongside other DDD methods to provide classification measures required for the detection process. This is mainly to enable accurate prediction of the driver's drowsiness state [17]. Furthermore, it is rare to find any DDD approach not utilizing at least one of these methods. However, they are highly vulnerable to deliberate false ratings or unintended bias [40].

TABLE. I. DDD APPROACHES AND DROWSINESS DETECTION CATEGORIES MATCHING

Categories Methods	Subjective	Physiological	Behavioral	Vehicle-based
Technological	-	x	x	x
Non-technological	x	-	-	-
Intrusive	-	x	x	-
Non-intrusive		-	x	x
Objective	-	x	x	x
Subjective	x	-	-	-

TABLE. II. STANFORD SLEEPINESS SCALE (SSS) [43]

Value	Description
1	Feeling active, vital, alert, or wide awake
2	Functioning at high levels, but not at peak; able to concentrate
3	Awake, but relaxed; responsive but not fully alert
4	Little foggy; not at peak
5	Foggy; losing interest in remaining awake; slowed down
6	Sleepy; woozy; fighting sleep; prefer to lie down
7	No longer fighting sleep; sleep onset soon; cannot stay awake

Stanford Sleepiness Scale [43] and Karolinska Sleepiness Scale [44] are the two most widely utilized subjective measures within the DDD domain. SSS is a 7-point measurement scale (Table II) describing the current state of drowsiness of an individual. This method as employed by [45] is most likely be used to categorize driver drowsiness into only two states. This is because of the close relation of each scale. KSS, on the other hand, is a 9-point scale (Table III). A contrast to SSS, this scale is considered a robust scale capable of categorizing driver's drowsiness into different levels [46]. Authors in [14] and [18] used KSS to define five, and three drowsiness states of their systems respectively. Also, a review of literature relating to DDD by [5] indicates KSS is the most preferred scale of these two scales.

TABLE. III. KAROLINSKA SLEEPINESS SCALE (KSS) [44]

Value	Sleepiness Level
1	Extremely alert
2	Very alert
3	Alert
4	Rather alert
5	Neither alert nor sleepy
6	Some signs of sleepiness
7	Sleepy but no difficulty staying awake
8	Sleepy with some effort to keep alert
9	Extremely sleepy, fighting sleep

B. Physiological Methods

Physiological methods are regarded as the most accurate DDD methods [9,47]. These methods provide access to the inner body state of the correspondent. With this capability, they can detect drowsiness at a very early stage [47]. Several sensors have been developed to track the electrical activities of different parts of the body. However, the currently employed (and most important) sensors for DDD are those involving heart activity, brain activity, and eyes activity [33]. The processes of capturing these body signals are termed electrocardiogram (ECG), electroencephalogram (EEG), and electrooculogram (EOG), respectively.

Physiological methods are conventionally intrusive [14,39]. The sensors used for capturing the required data are specific electrodes placed relative to the external part of the body they are tracking. For EEG, the electrodes are placed on the head [8]. The electrode is placed on the chest for ECG [27], and close to the eyes for EOG [1]. The intrusiveness of such methods has impeded their prospect of being the top DDD approach. Because of this, researchers are extensively exploiting new approaches, such as steering wheel movements (SWM) [17] and face tracking [16]. These approaches are quickly becoming the domain's point of interest.

Nevertheless, research advancements still continue within the physiological scope. These involve increasing the accuracy threshold [48], reducing the intrusiveness [8], reducing equipment's cost, and developing mobile solutions [49]. Author in [49] tried to balance intrusiveness, cost, and mobility by developing a low-cost EEG system using a wireless EEG headband and a smart watch, which still managed to get a reasonable accuracy. Below is a review of the three most prominent physiological methods:

- Electroencephalogram (EEG)

This method involves the tracking of electrical activities in the brain [30]. Of all physiological approaches, the EEG is the most widely used and most accurate [9,49]. Fig. 3 illustrates the flow diagram of such a system and its specifications [49]. [8] developed a wireless based brain-computer interface for detecting the driver's drowsiness. This system collects the driver's EEG signal at a sample rate of 256 Hz by installing the electrodes and a wireless transmitter (Bluetooth) on a brain cap to be worn by the driver.

- Electrocardiogram (ECG)

This method involves tracking the heart pulse rate [50]. Several types of researches have been carried out to facilitate this [12-13], and lots of comparisons have been done in order to prove its competence with other popular methods. Author in [13] examined the heart rate variability (HRV) and compared it with popular EEG and behavioral percentage of eyelid closure (PERCLOS) methods. Their results show that ECG predictions match the two approaches. Furthermore, with an aim to mitigate intrusiveness, [50] uses electrodes placed on the steering wheel and a wireless ECG sensor node to capture the driver's ECG signals through the palm. This setup is shown in Fig. 4. These methods, however, proved ECG to be a significant contender in the DDD space.

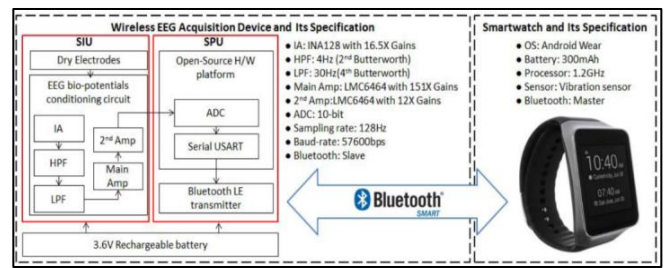


Fig. 3. Low-Cost and Mobile EEG System Diagram by [49].

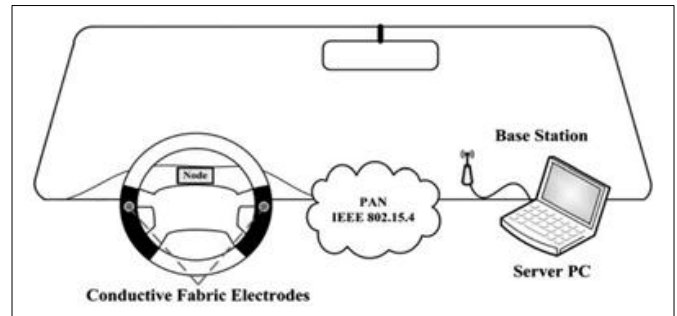


Fig. 4. ECG system architecture [50].

- Electrooculogram (EOG)

Like other physiological methods, EOG requires electrodes for data capturing. These electrodes, however, are used for tracking the electrical signals depicting the eyes' retina movements. Author in [1] conducted an EOG experiment with a sampling frequency of 512 Hz on 37 sleep-deprived subjects in a simulator. Although EOG signals are weaker than EEG signals, [47] proposed a system that was able to reach 80.74% accuracy.

C. Behavioral Methods

Unlike physiological methods, behavioral methods are non-intrusive [14]. They are capable of capturing the driver's drowsiness state without having any physical contact with the driver. These methods are the most explored DDD methods [27]. Percentage of eyelid closure (PERCLOS), a behavioral method, was one of the first and most commonly used DDD approach [5]. Rarely is there any DDD approach that does not consider PERCLOS. Other methods frequently use it as a threshold to validate their model [5,51].

As drivers begin to experience drowsiness, there are some physical changes that occur around their body- most, especially, around the head area [27]. These are the changes behavioral methods seek to track. Currently, the recognized changes within DDD scope are head nodding, yawning, and some varying eyelid states [27]. In order to detect these changes, behavioral methods usually go through a combination of video acquisition of the driver state and some computer vision techniques processes, [52-53] where computer vision encompasses both image processing and machine learning.

As stated above, the three main features targeted in behavioral methods are the head poses, yawning, and eye states. Several methods have been developed to determine either or a combination of these features. Author in [16] worked on a system tracking the eye state and head pose of the driver. Their

model defined three measures: head poses, eye index, and pupil activity. However, two of these measures are just part of the aforementioned eye state measure. Author in [15] also worked on eye states, and like [16], this measure was subdivided (into six separate measures). Other applications for yawning are also available and will be reviewed shortly.

While behavioral methods are currently popular, used in extensive research, these methods are usually impaired by individual specifics [14]. Many literature works realized this and mostly suggested that it tends to be too advanced for the time [15]. Author in [38] in an attempt to mitigate this issue included several drivers- specific thresholds like different eye shape, texture, and blinking patterns in their model. In [14], the author achieved theirs by incorporating several methods together. Other issues relating to behavioral methods include environment illumination, head rotation, and the cost and quality of the cameras used for video acquisition [26]. Below is a review of the three most prominent behavioral methods.

- Eye State Tracking

These methods involve making DDD decisions by extracting meaningful features from the driver's eye [54]. Features extracted are accumulated over a stipulated period of time so as to generate measurable variables for the actual drowsiness detection. Several measures have been defined and standardized within this context. These measures include PERCLOS, blink frequency, and velocity of eyelid opening and closing [5,27]. The blink frequency represents the number of times the eyelids close and open over a specific period of time. Fig. 5 represents a specific eye state interval for various eye state measures as part of a research in [15].

Lots of researches have been done with respect to the driver's eye state, thus it is regarded as one of the first and most accurate DDD approach [27]. Due to this, many comparisons are done against it. A comparative analysis done by [55], between eye closure and ECG (heartbeat signal), revealed that eye closure was more effective in detecting driver drowsiness than ECG. However, it is important to note that tracking the accuracy of eye states degrades, either when the driver is wearing glasses, in the presence of illumination or head position changes [27].

- Head Pose Estimation

This is a behavioral approach based on detecting whether the driver's head is nodding in a way similar to that of a drowsy person [17]. This process usually requires a 3D camera or a stereoscopic visioning. In the absence of a 3D camera, [16] used a 3D software (Blender) to create a 3D head model and continuously matched this with their 2D face sequence. Although it is highly probable for nodding to follow drowsiness, this is usually not the case as there are situations where drowsiness occurs without nodding, or probably, nodding occurs only when the drowsiness has been deeply infused [33]. In either of the two cases, it is evident that the accuracy and response time of the method would be degraded. However, this is recognized by researchers and this method is usually not used alone [16,56].

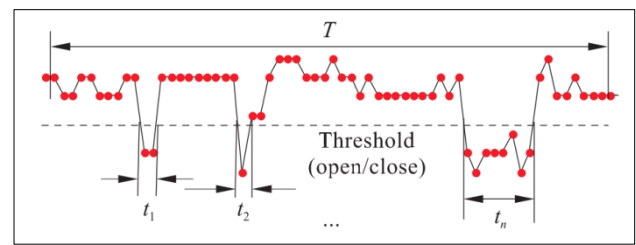


Fig. 5. Eye State Intervals for Measuring PERCLOS, and other Eye State Measures [15].

- Yawning Detection

This method involves tracking the driver's mouth [38]. However, like the head poses approach, this method does not always depict drowsiness [33]. Although this method is not usually used in isolation, [28,57] worked on systems solely for tracking the mouth and detecting driver's yawning.

- Vehicular Based Methods

These methods attempt to detect drivers' drowsiness by tracking changes in the behavior of the vehicles [5,27]. Several approaches have been developed to facilitate this, which includes steering wheel movements (SWM) [28,56], the standard deviation of lane position (SDLP) [59-60], and acceleration and deceleration fluctuations [27]. Of these three approaches, SWM and SDLP are the most popular [51,61]. Author in [17] worked on a low-cost SWM approach using a low-cost accelerometer and compared their results with other popular DDD approaches. Another implementation by [19] tracks the driver's lane deviation and then provides a warning to the driver to take action. However, if the driver does not respond to the warning, the system takes control and makes the correction itself.

Although vehicle-based methods have also recorded high accuracy rates, they are usually trailed by their high potential of false positive alarms. These have lead researchers to subject their systems to robust computations before reaching a reasonable accuracy. In order to reach the desired accuracy, [17] had to use several techniques and also compare their result with other methods. Many researchers, however, view this as a minor issue due to the computational power available today.

Conversely, SWM has been currently receiving much interest from researchers [5]. This is because of the possibilities of DDD based on the number of steering wheel corrections compared to normal driving conditions [27]. According to [5], many DDD solutions are currently targeting the development of low-cost SWM or the facial tracking approach. For SWM, [17] proposed a low-cost approach capable of obtaining a reasonable accuracy.

- Hybrid Methods

These methods involve the combination of any of the above DDD methods as a means for mitigating the weakness of each method when used individually [27]. These kinds of implementations are still at their infancy; however, the current implementations are indicating better results than other

methods used discretely [17]. In [14], the author developed a system using 23 measures including PERCLOS, SDLP and SWM which were captured through a driving simulator. Although their result did not attain the expected accuracy, their aim to consider individual specifics by correlating various approaches questions the results of current methods. Another

implementation by [62], that combined vehicle-based and behavioral approaches, were able to deduce that combination of more than one method yields higher accuracy.

Table IV summarizes and compares all current DDD methods available. It clearly states the techniques, accuracy and tools used in each of the methods.

TABLE. IV. COMPARATIVE ANALYSIS OF CURRENT DDD METHOD

Author(s)	Objective(s)	Techniques	Accuracy	Tools
[15]	- Address illumination and head posture changes issues with DDD based on facial features	- AdaBoost - ATM - ASM - JDS	- 86% overall accuracy	- Driver simulation setup - 8 bits RGB camera
[16]	- Drowsiness detection by analyzing driver's eye state and head pose	- SVM - Viola-Jones Algorithm - ATM - POSIT algorithm	- 97.2% on pupil detection - 87.27% overall accuracy	- BioID and Boston University database - Camera - Blender 3D software
[14]	- DDD model that considers drivers specifics - Validate MOL model by comparing with OL and ANN model	- MOL, OL, and ANN - KSS	- 3 DL: MOL has 64.15%, OL has 52.7%, ANN has 56.04% - 2 DL: MOL has 88.6%, ANN has 83.3%	- Simulator setup - SCANer studio software - Smarteye eye tracker and Pro software
[49]	- Develop a non-discrete DDD classification model	- EEG - SVM based Posterior Probabilistic Model (SVMPPM)	- 91.25% for alert - 83.78% for early-warning - 91.92% for full-warning	- Bluetooth Low-Energy - EEG headband with dry electrodes - Smartwatch - Matlab
[50]	- Measure driver fatigue and health condition - Provide a non-intrusive physiological means of DDD	- ECG - 100MHz sampling rate - HRV analysis	- not provided	- TinyOS - Zigbee - fabric electrodes - MCU TI MSP430
[13]	- Provide an HRV based system for detecting driver's fatigue early	- ECG, HRV - ANN - FFT	- 90% overall accuracy	- Matlab -
[8]	- Develop a wireless based brain-computer interface	- EEG - Clustering algorithm - 256Hz sampling rate	- 83.7% true positive rate (TPR)	- Braincap with EEG sensors - Bluetooth - 600 MHz processor
[17]	- develop a low-cost SWM DDD system suitable in real-world	- SWM, EEG, EOG, and PERCLOS - SVM	- 87.9% overall accuracy	- low-cost accelerometer - electrodes - camera
[47]	- Develop a wearable DDD system	- EOG - Sliding Window technique - Autoregressive Integrated Moving Average	- 0.5 seconds ahead of time alert rate	- EOG sensors - Arduino Uno - Android smartphone - HC-06 Bluetooth module
[63]	- develop an EOG classifier that can automatically detect microsleep	- EOG - 256 Hz sampling frequency - Sliding Window technique	- 57% precision - 93% recall	- EOG electrodes
[58]	- Detecting vehicles lane deviations using SWM	- SWM - Exponential Weighted Moving Average (EWMA) - Lane departure	- 91.24% accuracy	- accelerometer -

IV. METHODOLOGY

This section presents the system design details and its experimental setup. This design entails several machine learning models (which are used for drowsiness classification), alongside their evaluation measures. The machine learning models were evaluated in terms of their accuracy, sensitivity, specificity, false alarm rate, and miss rate. In addition, this section encompasses the data collection, data preprocessing, and feature selection approaches. These include a face and eye-state detection model and the calculation of the percentage of eyelid closure (PERCLOS), blink frequency (BF), and maximum closure duration (MCD) as the selected features.

A. System Design

As shown in Fig. 6, the operational design proposes a model following the conventional machine learning process. From this design, the first step is to acquire and preprocess the required data to be used for learning. This data was collected from the NTHU computer vision lab and preprocessed using several video and image processing techniques to detect the driver's eye states. The next step is to extract the targeted features, (from the preprocessed data) which are going to be used as the actual inputs for the learning process. The features extraction process was used to achieve this. This process helps in reducing the dimensionality of the raw inputs (i.e. each video frame pixels), and also in selecting meaningful variables through a combination of these raw inputs. Then, the drowsiness classification process was used to classify the driver's state into either an awake or drowsy state. This process utilizes several machine learning classification models. The result of these models is then evaluated for their accuracy, sensitivity, specificity, false alarm rate, and miss rate in the performance evaluation phase.

B. Dataset

The dataset used for this experiment was a driver drowsiness detection video dataset provided by the National Tsing Hua University (NTHU) Computer Vision Lab. The description of the dataset is given in Table V. This dataset consists of video data from both male and female drivers with various facial characteristics, different ethnicities, and from 5 different scenarios. The videos are in 640x480 pixels, 15/30 frames per second (fps) AVI format without sound. They were taken in real and varying illumination conditions. In addition to the video data, labels for each video frame were also provided whereby each frame was labeled with either a drowsy or a non-drowsy status.

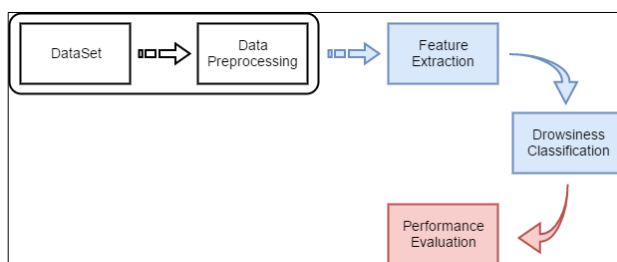


Fig. 6. Operational Design.

TABLE V. NTHU DATASET DESCRIPTION [64]

Dataset Category	Description
Training set	18 subjects with each subject having scenarios that contain each video type separately
Evaluation set	4 subjects with each subject having scenarios that combine all video types into one video
Scenario	Description
NoGlasses	Category for 30 frames per second videos and their respective labels
Glasses	
Sunglasses	
Night-NoGlasses	Category for 15 frames per second videos and their respective labels
Night-Glasses	
Driver's Behaviors	Description
Yawning	The driver opens his mouth wide due to tiredness
Nodding	The driver's head falls forward when drowsy or asleep
Looking aside	The driver turns his head left and right
Talking and laughing	The driver is talking or laughing while driving
Sleepy-eyes	The driver closes his eyes due to drowsiness while driving
Drowsy	The driver looks sleepy and lethargic (including nodding, slowly blinking and yawning)
Stillness	The driver drives normally
Videos	Description
yawning.avi	The video includes yawning behaviors
slowBlinkWithNodding.avi	The video includes sleepy-eyes and nodding behaviors
sleepyCombination.avi	The video includes a combination of drowsy behaviors, e.g. sleepy-eyes, yawning, nodding
nonsleepyCombination.avi	The videos include a combination of non-drowsy behaviors, e.g. laughing, talking, looking aside
Labels	Description
drowsiness.txt	0 for Stillness and 1 for Drowsy
head.txt	0 for Stillness, 1 for Nodding and 2 for Looking aside
mouth.txt	0 for Stillness, 1 for Yawning, and 2 for Talking or Laughing
eye.txt	0 for Stillness and 1 for Sleepy-eyes

C. Data Preprocessing

The raw data currently at hand is a 640x480 pixel 15/30 fps video collection and each frame's respective drowsiness annotation. If this data were to be used as it is, proposed machine learning classifiers would have 307,200 inputs/dimension values for each sample. This would have been very computationally expensive with a downside of defying the project aim seeking a low-cost solution, in the first place. Also, from the dataset, only data from 10 subjects in the training set were utilized for training while the whole evaluation set was used for the evaluation process. This totals to 192412 frames (1 hour, 46 minutes, 53 seconds). Thus, the dimension of the classifiers' input has to be reduced. This is what the preprocessing stage aims to achieve.

To reduce the input dimension, eye pupil detection is first performed on each frame by tracking the driver's face then eyes. This reduced the 307,200 inputs to just one value depicting the eye states. Then the eye state data was segmented for each video into a 4 secs segment collection. What this implies is that a 3 min video (i.e. 180 seconds) would have 45 segments, and each segment would contain 120 values, if 30 fps was used. The 4 secs segments are defined as the period (T) threshold and where drowsiness features are extracted from. The 4 secs time frame was selected because that duration is sufficient for experiencing a serious drowsiness damage [16]. And finally, the eye state data is normalized into an open or closed state because of the noises experienced and the capacity of the pupil eye detection model. These processes are further explained in the next sub-sections.

1) *Eye pupil detection*: The eye pupil detection model takes as input every video frame from the dataset. Then it converts this frame into a gray image, where several image processing operations are performed on it. Adaptive Histogram Equalization (AHE), Adaptive Thresholding, Morphological Transformations, face and eye detection using Haar Cascades, and Contours detection were the operations performed on each frame.

AHE was used to improve the contrast of the gray image. This was required to brighten the night video sets, and also to normalize the contrast of some of the daytime videos which were too light for detecting the driver's face or eyes. Then the driver's face and eyes were tracked using Intel's Haar face and eye cascades. However, only if the driver's face gets detected, the eyes can be tracked (within the detected face region.) This process can be visualized in Fig. 7.

Upon detecting the eye(s), adaptive thresholding was employed to binarize the image for the morphological operations. Erosion and Dilation are the basic morphological operations. Erosion erodes away the white (1) region in the image while dilation increases the white region. For this model, Closing (which is dilation followed by erosion) is performed, then Erosion, and finally Opening (which is erosion followed by dilation) on the binarized eye(s). The Opening was useful

for removing noises (scanty white pixels) in the eye image, and closing was used for closing small black points in the image. Then the eye pupil is detected by finding the area with the most concentrated black pixels (which is the eye pupil). This is defined as finding contours (continuous points having the same color or intensity) in OpenCV-Python documentation [65].

As shown in Fig. 7, the outputs can either be 0, 0.5, 0.75, or 1. The output is 0 if the face was detected but the eyes were not detected, or if the eyes were detected but the pupils were not detected. Also, the output is 0.5 if the face was not detected. This holds a value because detecting the face can be impeded by several behaviors such as nodding or looking aside, which are behaviors that are present in the dataset. And finally, the output is 0.75 if only one eye pupil is detected; otherwise, the value of the output is 1. A visualization of how the result looks like is shown in Fig. 8. From this figure, potential drowsiness measures can be identified, which can be exploited.

2) *Data segmentation*: The eye state data from the previous section and their associated labels were segmented into 4 secs segments before any further preprocessing. The 4 secs segment here is defined as 120 frames, which imply 120 eye states per segment. The 120 frames depict 4 secs because all the videos were later treated as 30 fps so as to enforce consistency in subsequent phases. Also, it is important to recall that every video has four associated label-files (drowsiness.txt, eye.txt, head.txt, and mouth.txt), thus every eye state segment also has four associated label-segments i.e. 480 labels for each eye state segment.

3) *Preprocessed data normalization*: Several causes, in addition to the presence of noise in the eye state data could be identified in the eye pupil detection section. Also, since only whether the eye is open or closed is tracked, the data has to be either 0 (drowsy) or 1 (awake). To achieve that, the head label of each eye state segment is used and converted the label from 1 (drowsy) to 0 and from 2 (looking aside) to 1. Then the remaining 0.75 values were converted to 1, and the 0.5s were converted to 1 or 0 depending on the statistical mode of the values 1 and 0 in the segment. The result of this process is visualized in Fig. 9.

4) *Labels combination*: As of now, each eye state segment is associated with four separate label segments, i.e., each eye state has four labels (drowsiness, eye, head, and mouth), and each of these labels could hold a different value. For example, head and mouth labels can contain a value 2 whereby drowsiness and eyes are just 0 and 1. Thus, these labels need to be aggregated into a single value depicting the drowsy state. To achieve this, all head and mouth labels were first converted with a value of 2 to 0 as these labels represent looking away and talking or laughing, respectively (both of which indicate awake states). Then, the average of the four labels was computed and binarized with a threshold of 0.25, i.e., everything greater than 0.25 is converted to 1 and 0 otherwise.

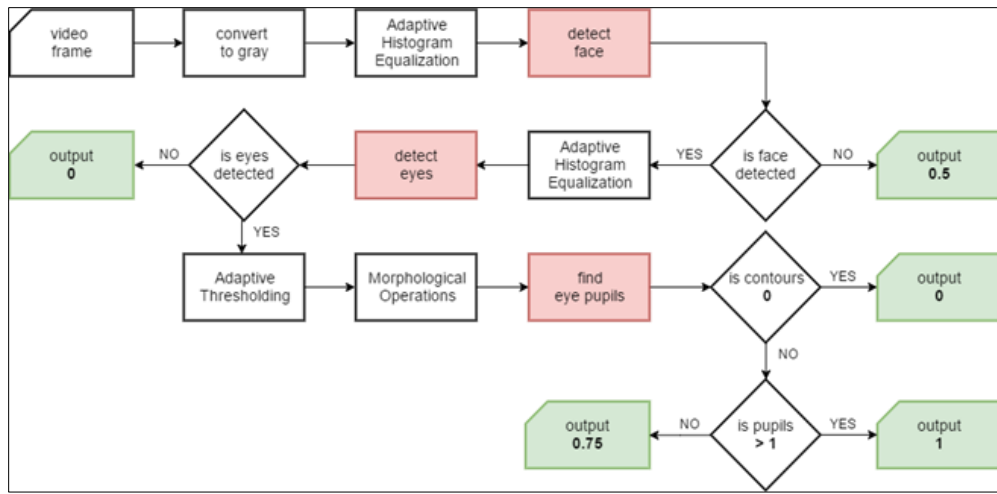


Fig. 7. Eye Pupil Detection Model.

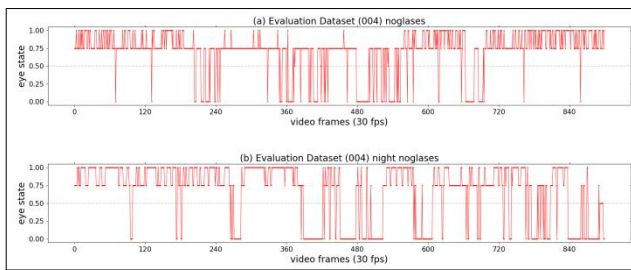


Fig. 8. Evaluation Dataset (004 Subject) Eye State.

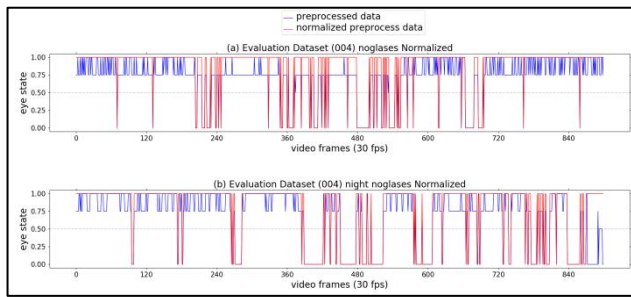


Fig. 9. Evaluation Dataset (004 Subject) Normalized Eye State.

D. Features Extraction

From the normalized eye state data, three features were extracted. This means that the features were calculated in every 4 secs within the video frame. The features calculated are the percentages of eyelid closure (PERCLOS), blink frequency (BF), and maximum closure duration (MCD). The calculation of these features is illustrated in Fig. 10.

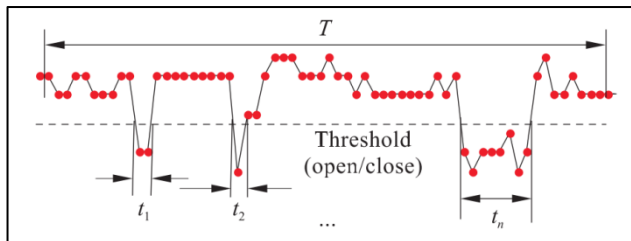


Fig. 10. PERCLOS, BF, and MCD Features Definition [15].

$$PERCLOS = \left(\frac{t_1 + t_2 + \dots + t_n}{T} \right) \times 100 \quad (1)$$

$$BF = \frac{n}{T} \quad (2)$$

$$MCD = \text{Max}(t_1, t_2, \dots, t_n) \quad (3)$$

where $T = 120$

1) *Features Standardization*: The next and final process after extracting the features was to scale the data so that their range is standardized. This process is required because the machine learning models calculate the distance between two samples using the Euclidean distance at some point in the learning process. Thus, if one feature has a broad range of values than the others, then the distance calculation would be greatly influenced by that feature. And for this data, the features were of the following range: 0 to 100 for PERCLOS because it's a percentage, 0 to 40 for BF, and MCD is 0 to 120 because the length of each data is 120, which is also the maximum amount of time a driver can have his/her eyes closed in a segment. These values indicate BF would not contribute much in the learning process.

There are two popular techniques for normalizing features in machine learning: Min-Max normalization (or rescaling), and Z-score normalization (or standardization). Min-Max rescales the features in the range [0, 1] or [-1, 1], while standardization rescales the features to have zero mean and a unit-variance. For this research, standardization was chosen because it is the most widely employed normalization technique for the range of machine learning algorithms used [66]. The equation for calculating this is shown below:

$$\text{standardization } \hat{x} = \frac{x - \bar{x}}{\sigma} \quad (4)$$

E. Drowsiness Classification

According to the DDD literature, most researchers usually scope their drowsiness classification to either an SVM or ANN machine learning model. This is logical as these two algorithms excel in high dimension, multi-class, and nonlinear classification problems. However, with a rigid data

preprocessing in place (as in this system), the capabilities of other algorithms can also be explored. Due to this, evaluating several KNN and logistic regression models was considered in the research. Also, the problem at hand is the binary classification problem. This is because the provided labels only identify a drowsy state (1) and an awake state (0).

1) *K-Nearest Neighbors (KNN) Models*: The KNN algorithm is one of the simplest and popular machine learning algorithms. This algorithm finds the nearest neighbors for a particular point in the sample space and (in a classification problem) then returns the class with the majority vote as the predicted output. For this research, 10 KNN models were designed with varying k values and used them for the classification problem. The details of this design are shown in Table VI.

$$Euclidean\ distance = \sqrt{\sum_{i=1}^k (x_i - y_i)^2} \quad (5)$$

2) *Support Vectors Machine (SVM) Models*: SVM is an algorithm that tries to find the optimal hyperplane that best separates data in an n-dimensional space. The terms optimal and best here indicate a hyperplane with the maximum margin from the support vectors. The support vectors are the points used to define the supporting hyperplane, which serve as the boundaries for the actual hyperplane. In order to find the optimal hyperplane, a kernel function is used to compute the similarity between other points and the support vectors, which is then further used to categorize the data points. The kernel functions extend SVM capability to higher dimensional spaces which thus makes it an important component in the algorithm. To exploit this capability, four are compared in Table VII.

$$linear: \langle x, \hat{x} \rangle \quad (6)$$

$$polynomial: (\gamma \langle x, \hat{x} \rangle + r)^d \quad (7)$$

$$RBF: (-\gamma \|x - \hat{x}\|^2) \quad (8)$$

$$sigmoid: (\tanh(\gamma \langle x, \hat{x} \rangle + r)) \quad (9)$$

where: $\gamma = \text{gamma}$, $d = \text{degree}$, and $r = \text{kernel projection}$

TABLE VI. KNN MODEL DESCRIPTION

Property	Description
k value	$k \in \{3, 5, 7, 9, 11, 13, 15, 21, 31, 41\}$
Distance metrics	Euclidean distance

TABLE VII. SVM MODELS DESCRIPTION

Property	Description
Kernel function K	$K(x, \hat{x}) \in \{linear, RBF, polynomial, sigmoid\}$
Distance metrics	Euclidean distance

3) *Logistic Regression Models*: Logistic Regression is a machine learning algorithm used for finding the best fitting boundary in a data. Unlike the name stipulates, this algorithm is used for binary classification. However, in order to evaluate its classification capability on the data, 6 different models were designed focusing on the algorithm's optimizer for minimizing its loss function. The structure of this design is detailed in Table VIII.

4) *Artificial Neural Networks (ANN) Models*: The artificial neural network is a model motivated by the biological structure of the human brain. This model enables the definition of neurons and their categorization into several layers. The input and output layers are the basic building blocks of an ANN model. However, if an ANN model contains just these two layers, it is no more different than a logistic or softmax regression model (depending on the output layer). ANN starts to become distinctive with the inclusion of hidden layers in the neural network. With this structure at hand, shown in Fig. 11, several models can be developed because different hidden layers, number of neurons, activation function, loss optimization function, and learning algorithm can be employed.

However, for this research, a 3 hidden layer neural network was created, with softmax being used as the activation function in the last hidden layer. Two models were developed from this layer structure with a [10, 50, 20] and [3, 27, 9] hidden neurons per hidden layer. For evaluation purposes, 6 optimization functions for minimizing loss were also applied alongside the hidden layer structure described previously, thus making a total of 12 models. Details about the models are displayed in Table IX.

TABLE VIII. LOGISTIC REGRESSION MODELS DESCRIPTION

Property	Description
Optimizer	$\varphi(x) \in \{GradientDescent, Momentum, Adagrad, Adadelta, RMSprop, Adam\}$
Learning rate	$lr \in \{0.01, 0.01, 0.01, 1.0, 0.0001\}$ for each optimizer respectively

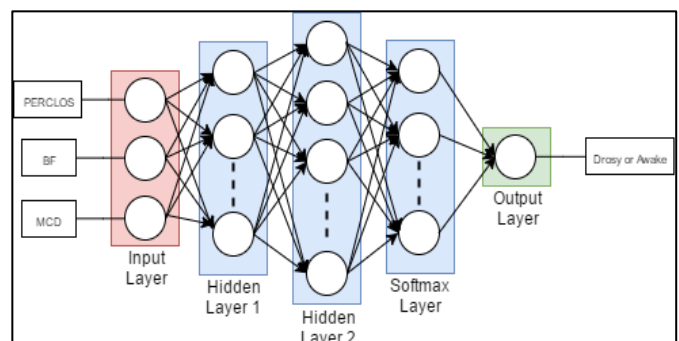


Fig. 11. ANN with 3 Hidden Model.

TABLE. IX. ANN MODELS DESCRIPTION

Property	Description
Optimizer	$\varphi(x)$ $\in \{GradientDescent, Momentum, Adagrad, Adadelta, RMSprop, Adam\}$
Learning rate	$lr \in \{0.2, 0.01, 0.1, 1.0, 0.0001\}$ for each optimizer respectively
Hidden layers and number of neurons	3 hidden layers with [10, 50, 20] and [3, 27, 9] for each optimizer.

F. Performance Evaluation

The final stage of the operational design is to evaluate the performance of the classification models used during the experiment. To do this, the conventional confusion matrix was used to generate the positive and negative prediction scores of each model and then deduct several performance measures from those scores. Details about the confusion matrix and the performance measures targeted are described in the next two sections. In addition to the confusion matrix performance measures, a 10-fold cross validation will also be performed on the KNN and SVM models.

1) *Performance Measures*: The confusion matrix is a contingency table used in machine learning for visualizing the raw performance of an algorithm. As shown in Table X, this matrix stipulates four scores which are true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The TP score depicts where the classifier correctly predicted a 1 or a truth value. TN, however, indicates where a 0 or false value was correctly predicted. The total of these two scores gives all the classifier’s correct predictions, from which the accuracy of the model can be calculated. The FP and FN values, on the other hand, provide wrong predictions, from which the model’s bad characteristics are found out in specific instances.

From the confusion matrix above, at least 9 performance metrics can be computed for the models. However, this paper would focus on just 5 of these measures. These metrics include accuracy, sensitivity (true positive rate), miss rate (false negative rate), false alarm rate (false positive rate), and specificity (true negative rate).

a) *Accuracy*: Accuracy is the basic performance measure for any model or system. The accuracy of a model depicts how close the predicted value is to the actual. In this context, this is how correctly the models predict a drowsy or awake state. The formula for calculating the models’ accuracy is shown below:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{10}$$

TABLE. X. CONFUSION MATRIX

		Prediction
Actual	True Positive (TP)	False Negative (FN) (type II errors miss)
	False Positive (FP) (type I error false alarm)	True Negative (TN)

b) *Sensitivity and Miss Rate*: Sensitivity measures the rate of positive prediction that is truly positive prediction. In other words, it refers to the proportion of predicted drowsiness states which may actually be drowsy. Furthermore, the complement of sensitivity is the miss rate. This is the proportion of actual drowsy states being predicted as awake states. In safety context, the miss rate is very crucial. In the coming section it may be observed that equal significance will be given to miss rate and accuracy when one attempts to determine the performance of the model. The formula below calculates the sensitivity and miss rate.

$$sensitivity = \frac{TP}{TP+FN} \tag{11}$$

$$miss\ rate = 1 - sensitivity \tag{12}$$

c) *Specificity and False Alarm Rate*: Like sensitivity and miss rate, specificity and false alarm rate measures the degree of negative predictions that are correctly and wrongly predicted. From the false alarm rate, the amount of distraction or discomfort of an actual system can be detected and mitigated. The formula for these two measures is stipulated below.

$$specificity = \frac{TN}{TN+FP} \tag{13}$$

$$false\ alarm\ rate = 1 - specificity \tag{14}$$

d) *Cross Validation*: As stated above, a 10-fold cross validation is performed on the KNN and SVM models. This is to confirm the generalization rate of these models so as to avoid overfitting. The cross validation was not performed on the logistic regression and ANN models because these models are very computationally intensive, thus a cross validation might consume significant time. And also, because the tensorflow library is used, it provides the capability to configure the number of steps for training and testing before converging; hence, it is possible to train or test for a long period of time before generating an output.

V. RESULTS AND DISCUSSION

The result of the standardized extracted features is shown in Fig. 12. As seen in the diagonal figures, the PERCLOS and MCD distribution explicitly depict the drowsiness states. This implies high PERCLOS and MCD should predict a drowsy state while their low values should predict an awake state. For BF, there is not much distinction between the awake and drowsy states. This is, however, expected because in the real-world, slow, medium, or fast blinks have an equal probability of being a drowsy or an awake state.

Finally, the correlation of each feature is also displayed in Table XI. These values and Fig. 12 indicate that BF has weak correlations with other features, with almost no correlation with PERCLOS. Again, putting this in a real-world context, the driver’s eyes could be closed for a long period of time within one to many blinks. Conversely, PERCLOS and MCD tend to have a strong positive correlation. This is because both features increase together. However, the scatters in the middle imply they do not decrease together. Sparse MCD values would total to a high PERCLOS.

TABLE. XI. CORRELATION BETWEEN FEATURES

Features	PC and BF	PC and MCD	BF and MCD
Correlation	-0.1709	0.8637	-0.4376

In this section, the performance result is presented for each model under their respective machine learning algorithm. A table will show these results and also accompany it with two figures visualizing the results based on two different comparison approaches. The first figure focuses on generic evaluation measures (like accuracy, sensitivity, and specificity) found in almost every machine learning analysis while the other figure focuses on measures (like miss rate and false alarm rate) that are critical and specific to safety contexts. Also, for evaluation, a benchmark of 70% accuracy and 15% miss rate is defined to determine how the good model should be.

A. K-Nearest Neighbors (KNN) Model Results

As seen in Table XII, all the models yielded varying results. However, they tend to have an average accuracy of 70.67 with $k = 3$ and $k = 31$ providing the least and best accuracy respectively. This can be further conceived by looking at Fig. 13, where it depicts that the accuracies increase as the k value increases, even though there were slight decreases when $k = 15$ and $k = 41$. This increment correlation can be justified because the models with a higher number of neighbors have more votes of the majority to aggregate from. Although, this can also mean overfitting, the notion is nullified by performing a 10-fold cross validation (CV) on the models. And as seen, the CV accuracies were very close to the original accuracies. This proves no overfitting occurred.

The model's sensitivity and specificity performance is shown in Fig. 13. These two metrics were very crucial in calculating the miss rate and false alarm rate. These two measures must be as high as possible so as to get low miss rate and false alarm rate values. The simulation reveals that $k = 31$ has the highest sensitivity value which also implies, it has the

lowest miss rate. This is convenient as this model had the best accuracy even though having high accuracy does not really signify having a high sensitivity or low miss rate. This is further asserted as this model ($k = 31$) does not have the best specificity and false alarm rate.

Fig. 14 compares the KNN model's miss rate and false alarm rate. From this figure, the miss rate and false alarm rate seems to decrease as the value of k increases. The best miss rates and false alarm rates were achieved at $k=31$ and $k=21$, respectively. Without this variation, $k=31$ would have been considered as a flawless model for the KNN algorithm. However, this is still tolerable as false alarm has a minimal safety effect on the driver. Thus, the best KNN model is when $k=31$.

B. Support Vector Machine (SVM) Model Results

For SVM evaluation, several kernel functions were compared. These functions transform the similarity computation between the input points and selected support vectors into different dimensions. The result of this experiment is shown in Table XIII. Like the KNN models, a 10-fold CV was performed to denounce overfitting. From the results, the sparseness of features is inferred when the linear model obtained higher accuracies than most of the non-linear models. This depicts the linearity between the features and the drowsiness states. This is further confirmed in the next section with logistic regression (a linear model), whereby the accuracies are between 69% and 70%.

The RBF model, however, performed better than the linear model. This is because the linear kernel is a special case of RBF [67] and with the RBF has the capability to fit cases (nonlinear categories), which the linear kernel cannot. The CV accuracies also showed no overfitting occurrence, except for the sigmoid function with a significant of 4% difference. Fig. 15 depicts the performance of different SVM models, briefly giving a bird's eye-view of their accuracies CV accuracies, sensitivities, and specificities.

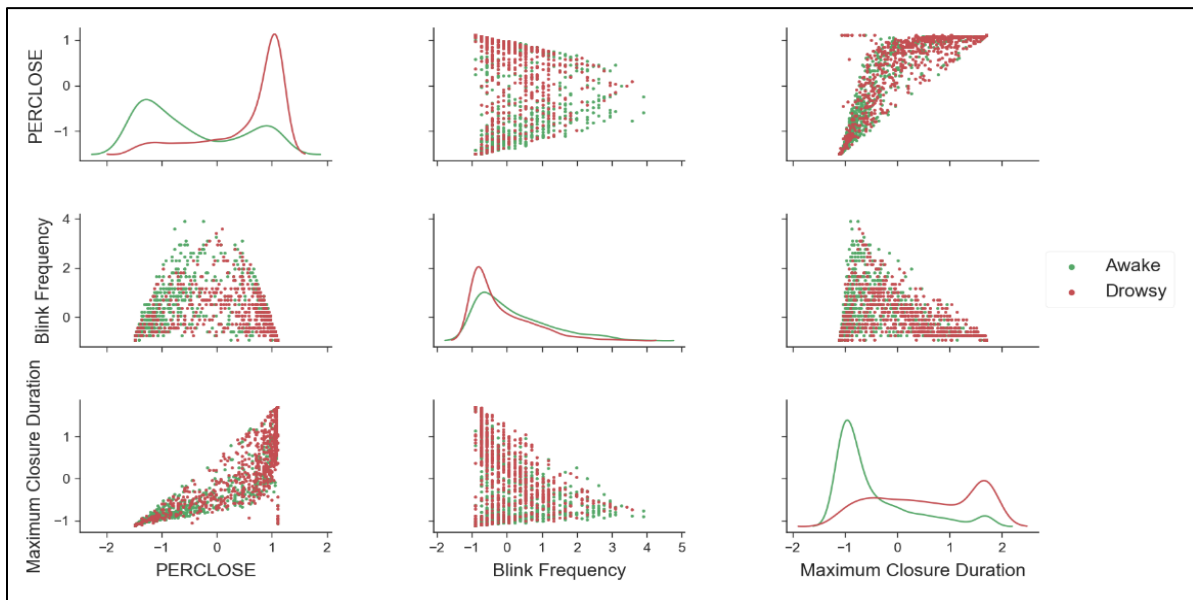


Fig. 12. Standardized Extracted Features.

TABLE. XII. KNN PERFORMANCE EVALUATION RESULTS

K value	Accuracy	CV accuracy	Sensitivity	Specificity	Miss rate	False alarm rate
3	67.78	64.11	76.67	55.81	23.33	44.2
5	69.38	67.46	79.17	56.18	20.83	43.82
7	69.06	68.42	79.17	55.43	20.83	44.57
9	70.81	68.42	82.22	55.43	17.78	44.57
11	71.29	68.90	82.5	56.18	17.5	43.82
13	71.45	69.06	82.78	56.18	17.22	43.82
15	70.97	68.10	82.5	55.43	17.5	44.57
21	71.93	69.06	81.94	58.47	18.06	41.57
31	72.25	70.33	83.33	57.30	16.67	42.7
41	71.77	71.29	81.94	58.05	18.06	41.95
Average	70.67	68.52	81.22	56.45	18.78	43.56

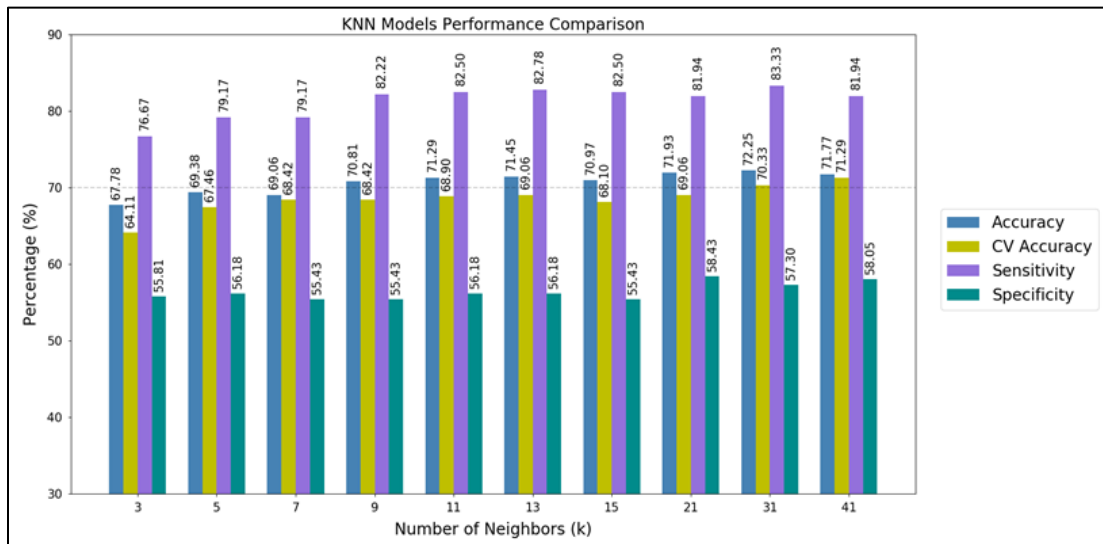


Fig. 13. KNN Models, Accuracy, CV Accuracy, Sensitivity, and Specificity.

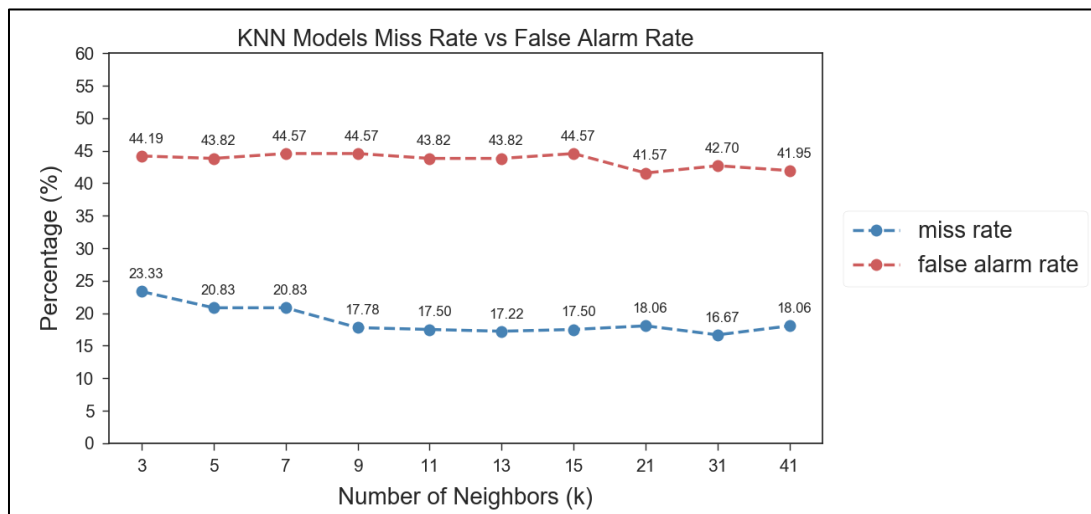


Fig. 14. KNN Models Miss Rate and False Alarm Rate.

TABLE. XIII. SVM PERFORMANCE EVALUATION RESULTS

Kernel function	Accuracy	CV accuracy	Sensitivity	Specificity	Miss rate	False alarm rate
Linear	69.86	68.42	78.61	58.05	21.39	41.95
RBF	70.66	69.7	80.56	57.30	19.44	42.7
Sigmoid	69.22	65.23	81.34	52.81	18.61	47.19
Polynomial	68.9	68.58	86.94	44.57	13.06	55.43
Average	69.66	67.98	81.86	53.18	18.13	46.82

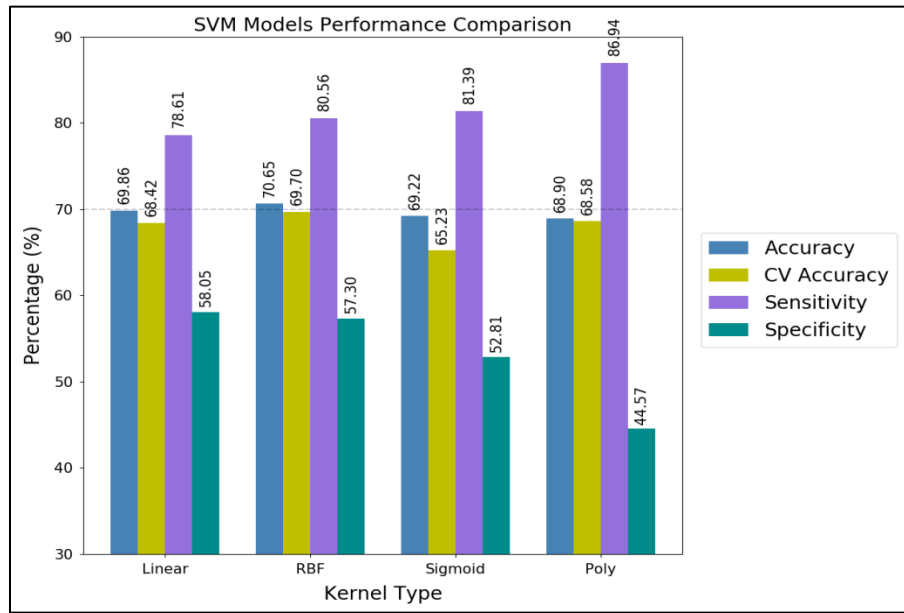


Fig. 15. SVM Models, Accuracy, CV Accuracy, Sensitivity, and Specificity.

In contrast to accuracy and specificity, the nonlinear kernels excelled at predicting the correct drowsy state (sensitivity) than the linear kernel. Thus, this means they have lower miss rate values. This is shown in Fig. 16, where the polynomial kernel had the lowest miss rate value. This is true because the strength of the polynomial kernel comes from it having more hyperparameters than the other kernels which thus increases its complexity. However, the RBF model was selected as the best model because it had the best accuracy, false alarm rate, and a considerable miss rate. The polynomial model on the other hand, had the worst accuracy and false alarm rate.

C. Logistic Regression Model Results

The logistic regression algorithm is a linear classifier for binary classification (awake or drowsy in this case). This algorithm tries to find the optimal weight required in fitting the best line through the data. In order to achieve this weight, an optimizer is required to minimize the loss function so that the predicted values are closer to their actual counterpart. Several models were developed to compare the result of different optimizers. The optimizers chosen were a category of optimizers developed as improvements to the popular gradient descent optimizer. Also, in order to achieve this, each optimizer's parameter was tuned so as to attain their possible

best result. The parameter values used are stipulated in Table XIV.

Table XIV, Fig. 17 and Fig. 18 show the result of each optimizer. As seen in the table and the figures depicted, Adam and gradient descent had the best accuracies. However, with respect to gradient descent, the two other optimizers, with 70% accuracy, are known for their great improvement on gradient descent's learning rate selection [68]. The Adam optimizer also had the best miss rate while RMSProp had the best false alarm rate. This is visualized in Fig. 18, where due to the linearity of the models, a (negative) correlation between the miss rate and false alarm rate was perceived. The Adam optimizer model was thus selected as the best model because this model obtained the best accuracy and miss rate.

D. Artificial Neural Network (ANN) Model Results

The final set of models evaluated was based on the ANN algorithm. The definitions of these models were similar to that of logistic regression, where different loss function optimizers were evaluated. However, since neural networks are being dealt with, two neural nets with 3 hidden layers were created whereby each hidden layer had different number of neurons. The result of this evaluation is shown in Table XV. As seen in this table, most of the models had 70% accuracy, thus signifying superiority to other machine learning algorithms compared with the provided data.

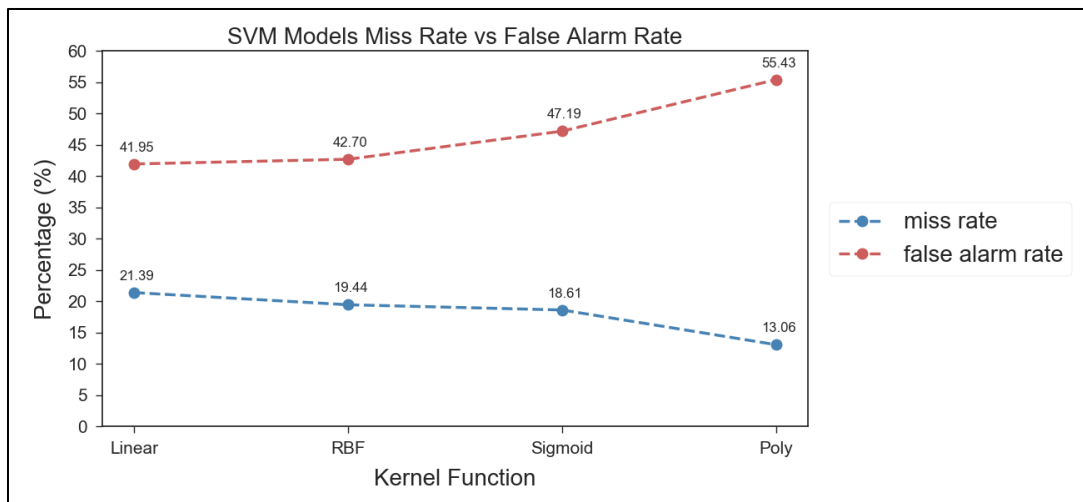


Fig. 16. SVM Models Miss Rate and False Alarm Rate.

TABLE. XIV. LOGISTIC REGRESSION PERFORMANCE EVALUATION RESULTS

Optimizer	Accuracy	Sensitivity	Specificity	Miss rate	False alarm rate
GradientDescent	70.81	81.11	56.93	18.89	43.07
Momentum	69.70	77.50	59.18	22.50	40.82
Adagrad	70.02	78.06	59.18	21.94	40.82
Adadelta	69.54	76.94	59.55	23.06	40.45
RMSprop	69.22	77.22	58.43	22.76	41.57
Adam	70.81	86.11	50.19	13.89	49.81
Average	70.02	79.49	57.24	20.51	42.76

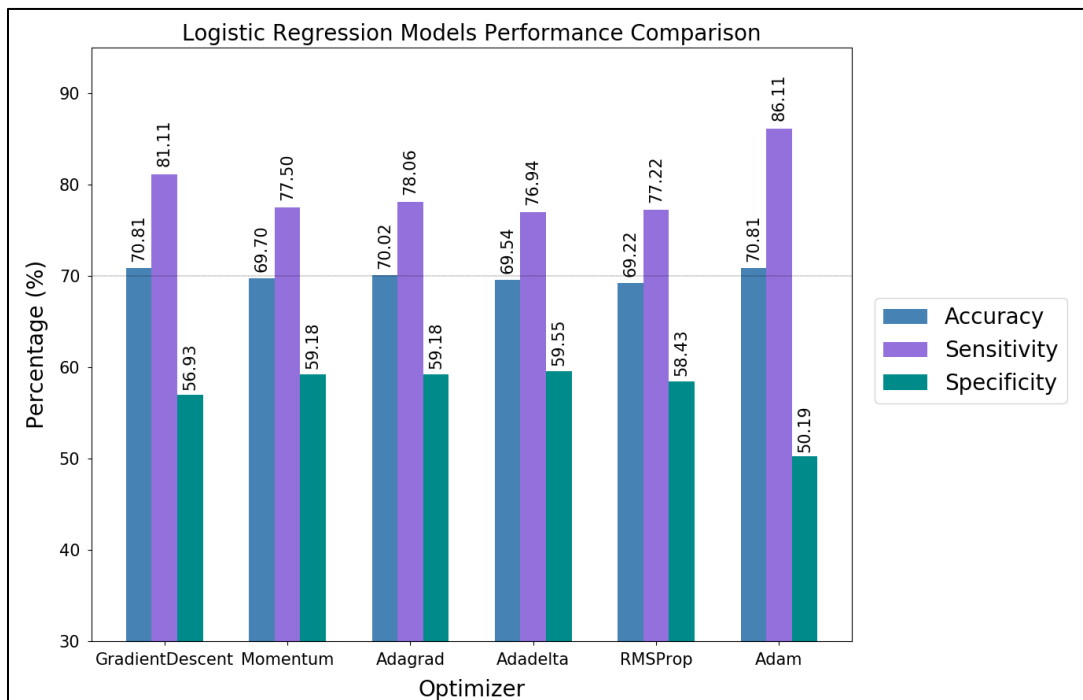


Fig. 17. Logistic Regression Models, Accuracy, Sensitivity, and Specificity.

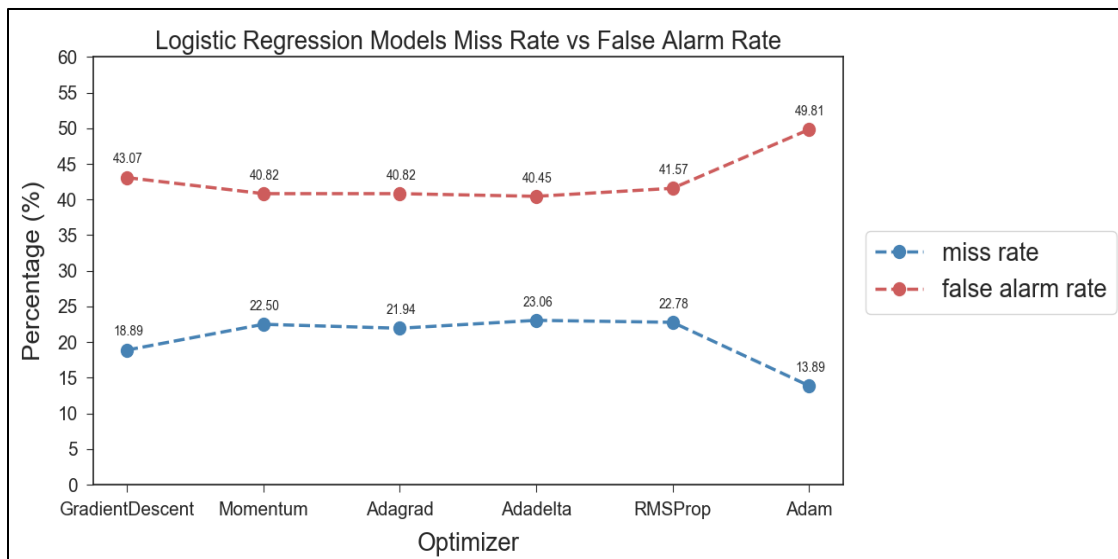


Fig. 18. Logistic Regression Models Miss Rate and False Alarm Rate.

TABLE. XV. ANN PERFORMANCE EVALUATION RESULTS

Optimizer/ Hidden layer	Accuracy	Sensitivity	Specificity	Miss rate	False alarm rate
GradientDescent[10, 50, 20]	71.13	81.67	56.93	18.33	43.07
Momentum[10, 50, 20]	71.45	85.28	52.81	14.72	47.19
Adagrad[10, 50, 20]	70.49	86.67	48.69	13.33	51.31
Adadelta[10, 50, 20]	70.34	86.39	48.69	13.61	51.31
RMSprop[10, 50, 20]	68.9	80.56	53.18	19.44	46.82
Adam[10, 50, 20]	71.61	85	53.56	15	46.44
Average	70.65	85	53.56	15	46.44
GradientDescent[3, 27, 9]	71.45	85	53.18	15	46.82
Momentum[3, 27, 9]	71.45	85.83	52.06	14.17	47.94
Adagrad[3, 27, 9]	69.7	78.06	58.43	21.94	41.57
Adadelta[3, 27, 9]	71.61	85.56	52.81	14.44	47.19
RMSprop[3, 27, 9]	71.45	85.83	52.06	14.17	47.94
Adam[3, 27, 9]	69.38	78.06	57.68	21.94	42.32
Average	70.84	83.06	54.37	16.94	45.63

Table XV has been translated to the graph depicted in Fig. 19 showing the comparison of performance measures of different ANN models. The results have been categorized based on the neural network structure. And as seen in the table, the network with 3, 27, and 9 neurons per layer obtained the best accuracy and false alarm rate, while the other network obtained the best miss rate. From this, it is inferred that the number of neurons per layer strongly affects these measures even though enough experimentation hasn't been carried out to generalize which network structure is most effective.

However, regarding the loss function optimizers, Adam and Adadelta attained the best accuracies in the [10, 50, 20] and [3, 27, 9] networks, respectively. The Adam and Adadelta optimizers were able to perform well in these scenarios because they computed adaptive learning rates for the traditional gradient descent parameters [68]. However, the

Adadelta was chosen in the neural network [3, 27, 9] as the best model for the ANN algorithm because its miss rate was better than that of Adam's as shown in Fig. 20.

E. Best KNN, SVM, Logistic Regression and ANN Model Comparisons

As shown in Fig. 21, each model has its strengths and weaknesses regarding accuracy, miss rate (compliment of sensitivity), and false alarm rate (compliment of specificity). The KNN model had the best accuracy and false alarm rate while competing with the SVM's false alarm rate. This is expected as KNN is a very good classifier when supplied clean data. However, the SVM model does not perform much compared to other models. This was surprising as SVM is a very strong classifier. This might be due to the parameter settings for the models. Thus, providing better or optimal parameters might improve the SVM performance. Nevertheless, this model had the best false alarm rate.

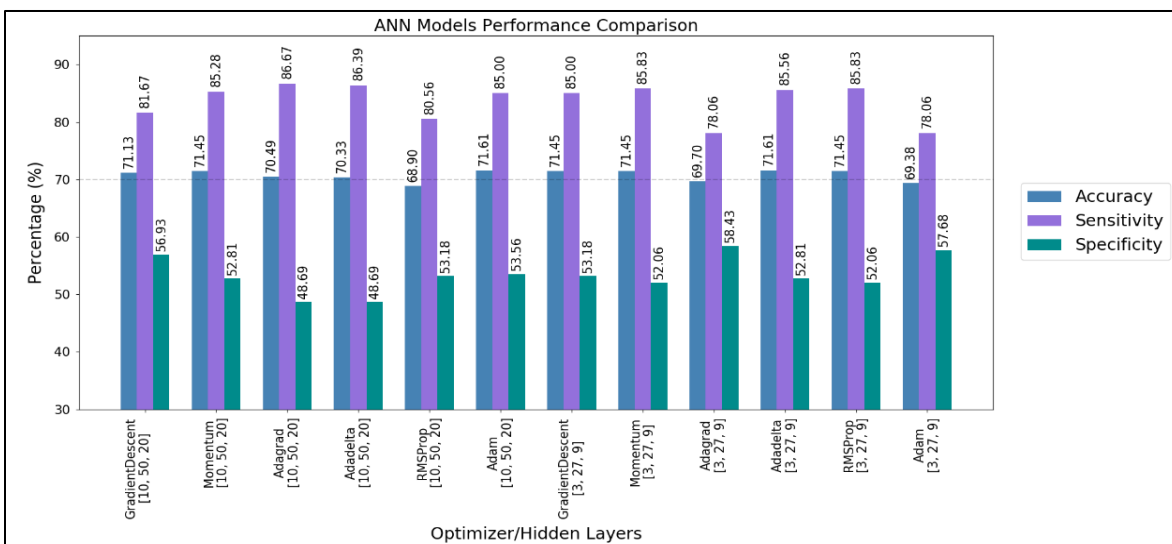


Fig. 19. ANN Models, Accuracy, Sensitivity, and Specificity.

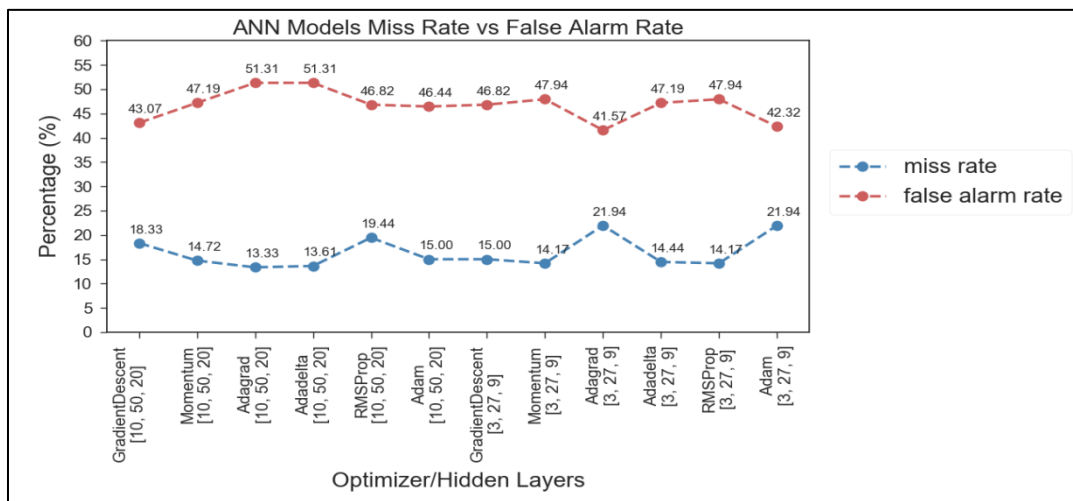


Fig. 20. ANN Models Miss Rate and False Alarm Rate.

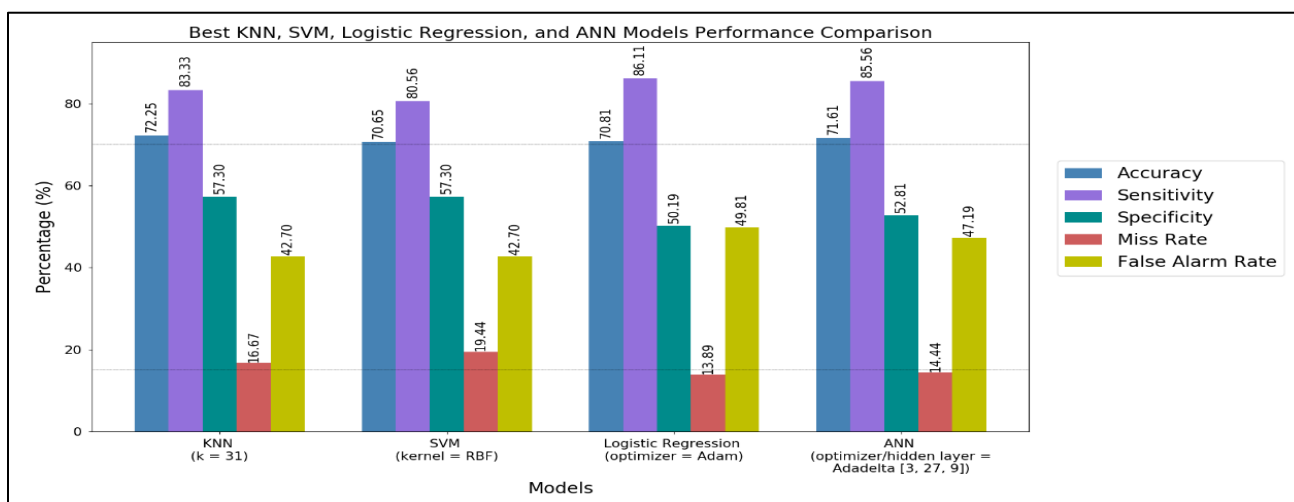


Fig. 21. Best KNN, SVM, Logistic Regression, and ANN Performance.

The logistic regression model had the best miss rate. This capability is very important as drowsy driving is considered a safety problem; thus, missing to classify a drowsy state as the awake state is a critical issue. Comparing this model to the KNN model, the KNN model performed better in all aspects except the miss rate where the logistic regression model surpasses it by 3%, this difference is still tolerable.

The final model is the ANN model. The result of this model can be interpreted to be better than the KNN model depending on the tradeoffs between accuracy and miss rate. However, this model obtained an accuracy of 71.61 and a miss rate of 14.44.

F. Result Benchmark

ANN and KNN results were compared with the paper recommended by the creator of the NTHU dataset. The result of this comparison is shown in Table XVI below. This paper by [69] provided two models using two different fusion strategies: independently-average architecture (IAA) and feature-fused architecture (FFA). And as shown below, the IAA and FFA obtained 73.06 and 70.81 accuracies, respectively. Author in [69] also evaluated the performance of popular multi-class classification algorithm on this dataset in their experiment. Comparing the result of this model and the other algorithms they evaluated with the result of proposed KNN and ANN model, proposed models obtained better accuracies than the latter, except their IAA model which had 73% accuracy.

TABLE. XVI. BENCHMARK RESULT

Researchers	Accuracy (%)
Proposed KNN	72.25
Proposed ANN	71.61
DDD-FFA [69]	70.81
DDD-IAA [69]	73.06
AlexNet [70]	62.99
VGG-FaceNet [71]	61.50
FlowImageNet [72]	67.85
LRCN [72]	65.93

VI. FUTURE WORK

As it was evident that the methods proposed in the research were non-intrusive and the results obtained were on par with the benchmarks considered, a real world prototype model will be developed. To start with, a simple prototype using a micro computer, capable of handling computer vision, like raspberry pi paired with a high resolution camera will be developed and tested. The output of the DDD system can be used to expand the research further or can be relayed to a control system of a vehicle to enable actuation based DDD output.

VII. CONCLUSION

This research focused on finding a low-cost and non-intrusive driver drowsiness detection solution which is based on face and eye tracking. To achieve this solution, several kinds of literature works were reviewed to understand the driver drowsiness detection ecosystem. These literature works

incited greater interest in this domain because driver drowsiness is considered as a major contributor to road accidents. With this problem at hand, it was discovered that most of the DDD researches, even though they obtained high accuracies, were unable to make it to real-world scenarios. With this research, this paper thus proposes a non-intrusive solution based on tracking the face and eyes of the driver, thus, not requiring any attachment to the driver. A low-cost solution was obtained by performing experimentation on a medium-quality video dataset. The eye pupil detection model and the numerous machine learning models were the contributions to this ecosystem.

However, the accuracy of the system was found to be at most 70-75%. In addition to this, the video feed required complex hardware and processing power, any lag due to which would result in fatal consequences. The video feed must be of high resolution and recorded in well-lit conditions, which may not be the case always in real world scenarios. This could lead to deviations in results. Any glares flashed, most importantly, on the driver's face might cause distractions or involuntary closure of eye lids, giving false alarms to the system. The data set must be acquired from various drivers not belonging to the same family, since genetic similarities can directly affect the specificity of the algorithms.

Although there are several improvements that can be embedded into this research, it is believed that the current result is adequate for deployment into a real system which can be used in the real-world scenario. This research is therefore considered as an inception for DDD ubiquity.

ACKNOWLEDGMENT

This work is supported by Universiti Teknologi Malaysia (UTM) and Ministry of Education (MOE) under grant no. 06G45, 17H58, 06G43 and 5F063. The authors would like to thank UTM and MOE for realizing and supporting this research work.

REFERENCES

- [1] Shuyan, H. & Gangtie, Z., 2009. Driver drowsiness detection with eyelid related parameters by Support Vector Machine. *Expert Systems with Applications*, 36(4), pp.7651-7658. Available at: <http://dx.doi.org/10.1016/j.eswa.2008.09.030>.
- [2] NHTSA, 2011. Traffic Safety Facts: Drowsy Driving. *National Highway Traffic Safety Administration*, 854(March), pp.1-3. Available at: <http://www-nrd.nhtsa.dot.gov/pubs/811449.pdf>.
- [3] World Health Organization, 2015. Global status report on road safety 2015. . pp.1-16.
- [4] Williamson, A. et al., 2014. Are drivers aware of sleepiness and increasing crash risk while driving? *Accident Analysis and Prevention*, 70, pp.225-234. Available at: <http://dx.doi.org/10.1016/j.aap.2014.04.007>.
- [5] Lenné, M.G. & Jacobs, E.E., 2016. Predicting drowsiness-related driving events: a review of recent research methods and future opportunities. *Theoretical Issues in Ergonomics Science*, (May), pp.1-21. Available at: <http://www.tandfonline.com/doi/full/10.1080/1463922X.2016.1155239>.
- [6] Komada, Y. et al., 2013. Short sleep duration , sleep disorders , and traffic accidents. *IATSSR*, 37(1), pp.1-7. Available at: <http://dx.doi.org/10.1016/j.iatssr.2013.06.001>.
- [7] Zhang, Y. & Hua, C., 2015. Optik Driver fatigue recognition based on facial expression analysis using local binary patterns. *Optik - International Journal for Light and Electron Optics*, 126(23), pp.4501-4505. Available at: <http://dx.doi.org/10.1016/j.ijleo.2015.08.185>.

- [8] Kumar, A., Chinara, S. & Sarkar, M., 2015. An application of wireless brain – computer interface for drowsiness detection. *Integrative Medicine Research*, 36(1), pp.276–284. Available at: <http://dx.doi.org/10.1016/j.bbe.2015.08.001>.
- [9] Johnson, R.R. et al., 2011. Drowsiness / alertness algorithm development and validation using synchronized EEG and cognitive performance to individualize a generalized model. *Biological Psychology*, 87(2), pp.241–250. Available at: <http://dx.doi.org/10.1016/j.biopsycho.2011.03.003>.
- [10] Li, G., Lee, B. & Chung, W., 2015. Smartwatch-Based Wearable EEG System for Driver Drowsiness Detection. , 15(12), pp.7169–7180.
- [11] Kranjec, J. et al., 2014. Novel methods for noncontact heart rate measurement: A feasibility study. *IEEE Transactions on Instrumentation and Measurement*, 63(4), pp.838–847.
- [12] Vicente, J. et al., 2011. Detection of driver's drowsiness by means of HRV analysis. 2011 Computing in Cardiology (CinC), pp.89–92. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6164509.
- [13] Patel, M. et al., 2011. Applying neural network analysis on heart rate variability data to assess driver fatigue. *Expert Systems with Applications*, 38(6), pp.7235–7242. Available at: <http://dx.doi.org/10.1016/j.eswa.2010.12.028>.
- [14] Wang, X. & Xu, C., 2016. Driver drowsiness detection based on non-intrusive metrics considering individual specifics. *Accident Analysis and Prevention*, 95, pp.350–357. Available at: <http://dx.doi.org/10.1016/j.aap.2015.09.002>.
- [15] Zhang, W., Cheng, B. & Lin, Y., 2012. Driver Drowsiness Recognition Based on Computer Vision Technology *, 17(3), pp.354–362.
- [16] Mbouna, R.O. et al., 2013. Visual Analysis of Eye State and Head Pose for Driver Alertness Monitoring. , 14(3), pp.1462–1469.
- [17] Lawoyin, S. et al., 2014. Detection methods for a low-cost accelerometer-based approach for driver drowsiness detection. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics, 2014-Janua(January)*, pp.1636–1641.
- [18] Lee, B., Lee, B. & Chung, W., 2016. Standalone Wearable Driver Drowsiness Detection System in a Smartwatch. , 16(13), pp.5444–5451.
- [19] Saito, Y., Itoh, M. & Inagaki, T., 2016. Driver Assistance System With a Dual Control Scheme: Effectiveness of Identifying Driver Drowsiness and Preventing Lane Departure Accidents. *IEEE Transactions on Human-Machine Systems*, 46(5), pp.660–671.
- [20] MIROS, 2016. Malaysian Institute of Road Safety Research (MIROS) | Road Facts. Available at: <https://www.miros.gov.my/1/page.php?id=17>.
- [21] FARS, 2015. FARS Acc _ Aux Analytical Users Manual The Tables in this Manual Contain Fatal Crash Counts Involving a Roadway Departure.
- [22] National Sleep Foundation, 2012. 2012 Sleep in America Poll: Planes, Trains, Automobiles, and Sleep. , (703), p.83.
- [23] Tefft, B.C., 2010. The Prevalence and Impact of Drowsy Driving. , pp.1–15. Available at: https://www.aaafoundation.org/sites/default/files/2010DrowsyDrivingReport_1.pdf.
- [24] Ingre, M. et al., 2006. Subjective sleepiness, simulated driving performance and blink duration: Examining individual differences. *Journal of Sleep Research*, 15(1), pp.47–53.
- [25] Johns, M.W. et al., 2008. A new method for assessing the risks of drowsiness while driving. *Somnologie - Schlafforschung und Schlafmedizin*, 12(1), pp.66–74. Available at: <http://www.scopus.com/inward/record.url?eid=2-s2.0-43549114043&partnerID=ZOTx3y1>.
- [26] Lawoyin, S.A., Fei, D.-Y. & Bai, O., 2014. A Novel Application of Inertial Measurement Units (IMUs) as Vehicular Technologies for Drowsy Driving Detection via Steering Wheel Movement. *Open Journal of Safety Science and Technology*, 04(04), pp.166–177. Available at: <http://www.scirp.org/journal/PaperDownload.aspx?DOI=10.4236/ojsst.2014.44018>.
- [27] Sahayadhas, A., Sundaraj, K. & Murugappan, M., 2012. Detecting driver drowsiness based on sensors: A review. *Sensors (Switzerland)*, 12(12), pp.16937–16953.
- [28] Saradadevi, M., 2008. Driver Fatigue Detection Using Mouth and Yawning Analysis. *International Journal of Computer Science and Network Security*, 8(6), pp.183–188.
- [29] Sabet, M. et al., 2012. A New System for Driver Drowsiness and Distraction Detection. , pp.1247–1251.
- [30] Chen, L. et al., 2015. Automatic detection of alertness / drowsiness from physiological signals using wavelet-based nonlinear features and machine learning. *EXPERT SYSTEMS WITH APPLICATIONS*, 42(21), pp.7344–7355. Available at: <http://dx.doi.org/10.1016/j.eswa.2015.05.028>.
- [31] Forsman, P.M. et al., 2013. Efficient driver drowsiness detection at moderate levels of drowsiness. *Accident Analysis and Prevention*, 50, pp.341–350. Available at: <http://dx.doi.org/10.1016/j.aap.2012.05.005>.
- [32] Williamson, A. et al., 2014. Are drivers aware of sleepiness and increasing crash risk while driving? *Accident Analysis and Prevention*, 70, pp.225–234. Available at: <http://dx.doi.org/10.1016/j.aap.2014.04.007>.
- [33] Chacon-Murguia, M.I. & Prieto-Resendiz, C., 2015. Detecting Driver Drowsiness A survey of system designs and technology. *IEEE Consumer Electronics Magazine*, (October).
- [34] Kati, B. & Steven, K., 2015. Drowsiness: Causes, Treatments & Prevention. Available at: <http://www.healthline.com/symptom/drowsiness> [Accessed December 5, 2016].
- [35] Czeisler, C.A. et al., 2016. Sleep-deprived motor vehicle operators are unfit to drive: a multidisciplinary expert consensus statement on drowsy driving ☆. *SLEH*, 2(2), pp.94–99. Available at: <http://dx.doi.org/10.1016/j.sleh.2016.04.003>.
- [36] Mahachandra, M. & Garnaby, E.D., 2015. The effectiveness of in-vehicle peppermint fragrance to maintain car driver ' s alertness. *Procedia Manufacturing*, 4(Iess), pp.471–477. Available at: <http://dx.doi.org/10.1016/j.promfg.2015.11.064>.
- [37] Johns, M.W. et al., 2008. A new method for assessing the risks of drowsiness while driving. *Somnologie - Schlafforschung und Schlafmedizin*, 12(1), pp.66–74. Available at: <http://www.scopus.com/inward/record.url?eid=2-s2.0-43549114043&partnerID=ZOTx3y1>.
- [38] Jo, J. et al., 2014. Detecting driver drowsiness using feature-level fusion and user-specific classification. *Expert Systems With Applications*, 41(4), pp.1139–1152. Available at: <http://dx.doi.org/10.1016/j.eswa.2013.07.108>.
- [39] Chung, B.L.S.J.W., 2011. Real-time physiological and vision monitoring of vehicle driver for non-intrusive drowsiness detection. , (May), pp.2461–2469.
- [40] Shahid, A., Shen, J. & Shapiro, C.M., 2010. Measurements of sleepiness and fatigue. *Journal of Psychosomatic Research*, 69(1), pp.81–89. Available at: <http://dx.doi.org/10.1016/j.jpsychores.2010.04.001>.
- [41] Shen, J., Barbera, J. & Shapiro, C.M., 2006. Distinguishing sleepiness and fatigue: Focus on definition and measurement. *Sleep Medicine Reviews*, 10(1), pp.63–76.
- [42] Čolić, A., Marques, O. & Furht, B., 2014. Driver Drowsiness Detection and Measurement Methods. In *Driver Drowsiness Detection: Systems and Solutions*. Cham: Springer International Publishing, pp. 7–18. Available at: http://dx.doi.org/10.1007/978-3-319-11535-1_2.
- [43] Hoddes, E. et al., 1973. Quantification of Sleepiness: A New Approach. *Psychophysiology*, 10(4).
- [44] Åkerstedt, T. & Gillberg, M., 1990. Subjective and objective sleepiness in the active individual. *International Journal of Neuroscience*, 52(1-2).
- [45] Mcdonald, A.D., 2014. Improving Driver Drowsiness Detection through Temporal, Contextual, and Hierarchical Modeling By.
- [46] Putilov, A.A. & Donskaya, O.G., 2013. Construction and validation of the EEG analogues of the Karolinska sleepiness scale based on the Karolinska drowsiness test. *Clinical Neurophysiology*, 124(7), pp.1346–1352. Available at: <http://dx.doi.org/10.1016/j.clinph.2013.01.018>.
- [47] Ma, Z., Li, B.C. & Yan, Z., 2016. Wearable driver drowsiness detection using electrooculography signal. *WiSNet 2016 - Proceedings, 2016 IEEE Topical Conference on Wireless Sensors and Sensor Networks*, pp.41–43.

- [48] Yeo, M.V.M. et al., 2009. Can SVM be used for automatic EEG detection of drowsiness during car driving? *Safety Science*, 47(1), pp.115–124. Available at: <http://dx.doi.org/10.1016/j.ssci.2008.01.007>.
- [49] Li, G., Lee, B. & Chung, W., 2015. Smartwatch-Based Wearable EEG System for Driver Drowsiness Detection. , 15(12), pp.7169–7180.
- [50] Jung, S., Shin, H. & Chung, W., 2014. Driver fatigue and drowsiness monitoring system with embedded electrocardiogram sensor on steering wheel. *IET Intelligent Transport Systems*, 8(1), pp.43–50.
- [51] Loon, R.J. Van, Brouwer, R.F.T. & Martens, M.H., 2015. Drowsy drivers' under-performance in lateral control: How much is too much? Using an integrated measure of lateral control to quantify safe lateral driving. *Accident Analysis and Prevention*, 84, pp.134–143. Available at: <http://dx.doi.org/10.1016/j.aap.2015.08.012>.
- [52] Boguslaw, C. & Sławomir, G., 2014. Neurocomputing Hybrid computer vision system for drivers' eye recognition and fatigue monitoring Bogusław Cyganek n , Sławomir Gruszczyński , 126, pp.78–94.
- [53] Fogelton, A. & Benesova, W., 2016. Eye blink detection based on motion vectors analysis. *Computer Vision and Image Understanding*, 148, pp.23–33. Available at: <http://dx.doi.org/10.1016/j.cviu.2016.03.011>.
- [54] Zhou, T., Wang, S. & Cheng, D., 2013. Design and implementation of a real-time eye tracking system. *The Journal of China Universities of Posts and Telecommunications*, 20(August), pp.1–5. Available at: [http://dx.doi.org/10.1016/S1005-8885\(13\)60260-5](http://dx.doi.org/10.1016/S1005-8885(13)60260-5).
- [55] Osuga, S. & Hiraki, K., 2013. Physiological and accident prevention effects of sleep alert systems: Comparative study of eye closure and heart rate based alarms. In 20th ITS World Congress Tokyo 2013.
- [56] Teyeb, I. et al., 2013. A Novel Approach for Drowsy Driver Detection Using Head Posture Estimation and Eyes Recognition System Based on Wavelet Network.
- [57] Shabnam, A., 2012. Driver Drowsiness Monitoring Based On Yawning Detection,
- [58] Assuncao, A.N., Paula, F.O. De & Oliveira, R.A.R., 2015. Methodology to Events Identification in Vehicles Using Statistical Process Control on Steering Wheel Data. , pp.1–4.
- [59] Imai, A. & Oguri, K., 2013. Estimation of driver's drowsiness level considering a characteristic sleepiness transition of drowsy driving. In 20th ITS World
- [60] McDonald, a. D. et al., 2013. Steering in a Random Forest: Ensemble Learning for Detecting Drowsiness-Related Lane Departures. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 56(5), pp.986–998. Available at: <http://hfs.sagepub.com/cgi/doi/10.1177/0018720813515272>.
- [61] Morris, D.M., Pilcher, J.J. & Switzer, F.S., 2015. Lane heading difference: An innovative model for drowsy driving detection using retrospective analysis around curves. *Accident Analysis and Prevention*, 80, pp.117–124. Available at: <http://dx.doi.org/10.1016/j.aap.2015.04.007>.
- [62] Cheng, B. et al., 2012. Driver Drowsiness Detection Based on Multisource Information. *Human Factors and Ergonomics in Manufacturing*, 16(1), pp.61–81. Available at: <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Human+Motion+Simulation+for+Vehicle+and+Workplace+Design#1>.
- [63] Hutchison, D. & Mitchell, J.C., 2012. Information Technology in Bio-and Medical Informatics. Imp004, 7451, pp.123–132. Available at: <file:///C:/Users/matte/AppData/Local/Mendeley Ltd./Mendeley Desktop/Downloaded/Hutchison, Mitchell - 2006 - Lecture Notes in Computer Science.pdf><http://link.springer.com/10.1007/978-3-642-32395-9>.
- [64] NTHU, 2016. NTHU Driver Drowsiness Dataset Readme. Available at: ftp://140.114.77.242/Training_Evaluation_Dataset/README.txt.
- [65] Alexander, M. & K., A., 2013. OpenCV-Python Contours. Available at: http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_contours/py_contours_begin/py_contours_begin.html#contours-getting-started.
- [66] Grus, J., 2015. Data Science from Scratch, Sebastopol: O'Reilly.
- [67] Hsu, C., Chang, C. & Lin, C., 2016. A Practical Guide to Support Vector Classification. , 1(1), pp.1–16.
- [68] Ruder, S., 2016. An overview of gradient descent optimization. , pp.1–12.
- [69] B, S.P. et al., 2017. Driver Drowsiness Detection System Based on Feature Representation Learning Using Various Deep Networks., pp.154–164.
- [70] Krizhevsky, A. & Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks. , pp.1–9.
- [71] Parkhi, O.M., Vedaldi, A. & Zisserman, A., 2015. Deep Face Recognition. , (Section 3).
- [72] Donahue, J. et al., 2015. Long-term Recurrent Convolutional Networks for Visual Recognition and Description.